ADMIN Network & Security

Automation | AD RMS | Cookie Dangers

ADMIN

Network & Security

ISSUE 89

Automation

- Microsoft Power Automate
- Ansible Automation Platform

DNS Configuration Errors

AzureADRecon

Find Entra ID vulnerabilities

Monitoring IoT Devices

DRBD9 and DRBD Reactor

Create a high availability VM

Cookie Dangers

LINUX NEW MEDIA
The Pulse of Open Source

AD RMS
A built-in server role
to prevent data loss

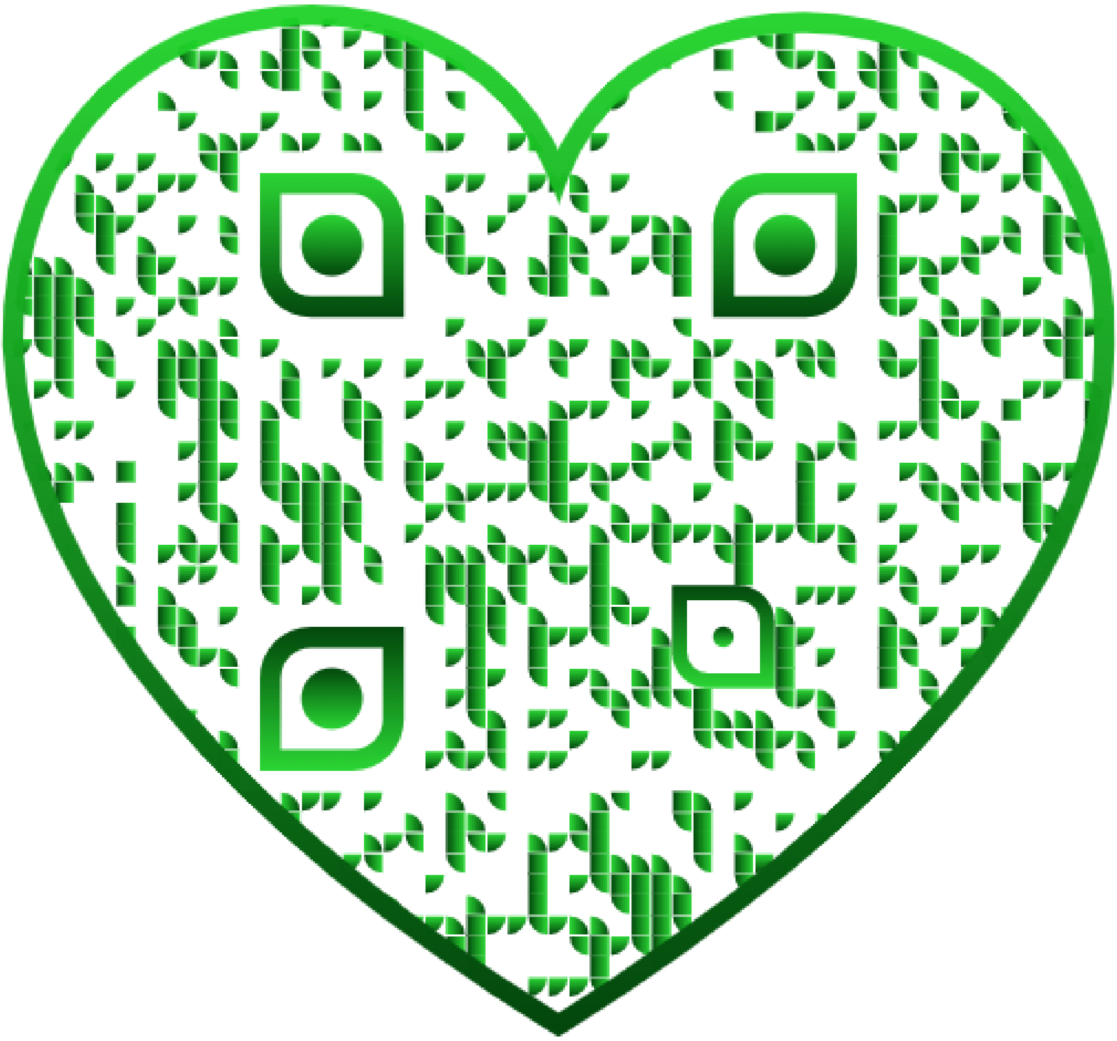
Goss
Run checks on Linux
and Windows computers

Falco
Proactive threat
detection for containers

Alerta
Aggregate alerts from
multiple monitoring systems

IPFire_
2.29
ISSUE 89/2025
ADMIN
Network & Security
DVD INSIDE

WWW.ADMIN-MAGAZINE.COM



Performance From Another Star

TUXEDO Stellaris 16 - Gen7

Quiet high-performance workstation or desktop performance in a portable format. The Stellaris 16 combines the fastest processors and graphics cards on the market with a quiet and powerful cooling system in a portable very sturdy all-aluminum chassis for a galactical balance of performance, fan acoustic comfort and mobility.

Flanked by up to 128 GB RAM and ultra fast PCIe 5.0 SSD storage, the Linux gaming workstation offers performance from another star, while the max-size 99 Wh battery allows for over 8 hours of outdoor expeditions and the HDR Mini LED display raises the image quality to extra terrestrial heights.

Linux compatible

Up to 5 Years Guarantee

Immediately ready for use

TUXEDO

tuxe.do/lxadmin89

Made in Germany

German Data Privacy

German Tech Support

It's Never Too Late for the Truth

Who knew my primary mission was to suffer fools gladly?

Much like any customer-oriented job, system administration is a service industry position. Yes, you're an IT professional, but customer service is your real job. I know that we all believe, falsely, that we work with computers. We tell the world, "I'm a computer person," but in reality, "You're a people person." We are no different from restaurant waitstaff. We play a roll. We deliver service. We do things behind the scenes that one hopes will make our customers (users) happy and more efficient. Our job is to please them. Don't roll your eyes or sigh: Allow me to recount an experience before you turn the page and look for something more technical to read.

I recently related a story to one of my colleagues that described a situation where I argued (belabored) my point about a particular decision I felt was not only incorrect, but downright detrimental. I punctuated my brilliant story with, "I don't suffer fools gladly." He stared into the great beyond before looking at me and wisely retorted, "You know, Ken, only about 1 percent of what you're paid for is your subject matter knowledge and skill set. The other 99 percent of what you're paid for is to suffer fools gladly."

I laughed.

After parting ways, I thought about his words with discernment. I pondered the concept at some length and concluded that his assertion is 1,000 percent correct. I looked back on my career and my many personnel conflicts over the years and that little imaginary light bulb went on over my head. Could it be that this is my 42 and that the answer to the ultimate question of my purpose in life as a system administrator is to suffer fools gladly? I was both shocked and intrigued. I, surprisingly, was speechless at the whole scenario that had just played out in my head.

I realized for the first time that I'm not really a computer guy at all – I'm a customer caregiver. I'm merely the food server between the kitchen (data center) and the hungry customer (end user). This realization didn't put me into the tailspin you might have assumed it would. In fact, I took a deep breath and decided that, had I known my actual roll 30 years ago, my life and my career would have gone so much differently than it has. I've had a good and productive career, but I would have had fewer conflicts and a happier existence if I'd had this conversation in my early days.

This epiphany has taken a huge weight off me. I never knew how high-strung I was until I allowed myself to relax and enjoy a day at work. The tension in my jaws released. I never noticed that I kept my stomach in knots until I didn't. I've been a coiled snake ready to strike for most of my days – fangs out waiting for a victim to trip my snare or enter my domain uninvited with a dumb question or silly request. I was the rubber band, stretched too tight, waiting to snap at the first sign of some unreasonable rule or ridiculous new hoop to jump through.

Sure, it might just be me. You might never have experienced these feelings. If not, congratulations for having figured it out for yourself. If you have, I hope this story helps you as much as it has me. One of the positive attributes is that I listen to wisdom and glean new behaviors from it. I try to learn from others' mistakes so that I don't have to repeat them. Conflicts are bound to happen, no matter what you do or how wise you are. That's just human nature. How you handle those conflicts is the difference between living a peaceful, productive life and living one on the edge of despair. Being perpetually angry and uptight is not a good look for anyone, and it makes for an exhausting existence. Remember that no matter how much you know, how adept you are at typing commands or rebooting computers, suffering fools gladly is what you're paid to do. Everything else is just screen time.

Ken Hess • Senior ADMIN Editor

Lead Image © vska, 123RF.com

Features

10 Microsoft Power Automate
The MSPA platform for automating work processes uses a wide range of connectors to integrate seamlessly with the Microsoft environment.

14 Ansible Automation Platform
If you need a management layer to wrap around Ansible, are looking to build pipelines, or simply need greater insights into the status of your automation, AAP offers useful features.

Service

3 Welcome
6 News
96 Back Issues
97 Call for Papers
98 Coming Next Month

Tools

20 Alerta
This free software fields alerts from several monitoring systems and delivers them in the web GUI or at the command line.

26 DRBD9 and DRBD Reactor
Create a Linux high-availability stack for virtual instances with replicated storage comparable to the classic Corosync and Pacemaker solution.

32 Goss
Get started with this free tool that checks your servers, delivers a machine health status report, and evaluates test results on Linux and Windows computers.

36 WSUS Deprecation
With the discontinuation and deprecation of the Windows Server Update Services at the end of Windows 2025 support, you should know whether you need to take action now and what tools are available as alternatives for Windows updates and patch distribution.

Containers and Virtualization

40 AzureADRecon
Identify vulnerabilities at an early stage in Entra ID environments.

44 Falco
Detect threats with the use of a driver that generates a data stream from events in containerized applications.

Security

50 AD Rights Management
A built-in Microsoft AD RMS server role can help prevent data loss in your organization.

54 Cookie Dangers
Session data stored on user hard disks can be targeted by criminals to bypass the need for a password or second factor in active user sessions.

56 Hardening Linux Servers
Delve into the software treasure chest to find useful products that improve virus protection on Linux.

62 Trivy
Take a deep dive into vulnerabilities, misconfigurations, and secrets with guidance on performance tuning, security configurations, and scalability across Linux distributions.

Code and listings for *ADMIN* can be found here:
https://linuxnewmedia.thegood.cloud/s/9nFQcFb2p8oRMEJ

10, 14 | Automation

- **Microsoft Power Automate**
- **Ansible Automation Platform**

Optimize, automate, and manage workflows and processes in your data center.

Highlights

20 Alerta
Aggregate alerts from various monitoring systems to determine what environments are operating normally and where things are currently falling apart.

40 AzureADRecon
Analyze Entra ID environments, generate reports, and identify potential risks at an early stage by providing detailed insights into users, roles, and service principals.

76 PS Module Repos
Create efficient and reusable code by abstracting PowerShell scripts into functions bundled into modules and made available in a central repository for improved automation processes.

Management

66 F/LOSS Lifecycle Management
Many proprietary tools on the market are designed to automate server management in modern data centers, but we look at the tools offered by the open source community.

71 IoT Monitoring
Keeping track of a large number of Internet of Things devices means understanding IoT networks, developing suitable monitoring metrics, and (ideally) integrating the IoT with infrastructure monitoring.

@adminmag

ADMIN magazine

@admin-magazine.com

@adminmagazine@hachyderm.io

Nuts and Bolts

76 Own Repos for PS Modules
Optimize the way you organize and maintain scripts by bundling PowerShell modules and deploying them internally.

80 DNS Config Errors
Correct name resolution is a fundamental prerequisite for trouble-free communication between servers and between servers and users. We look at practical tools for troubleshooting DNS configurations.

86 GnuPG
We look at certificate management with a web key directory and a directory service and investigate automated authentication with GnuPG actium.

92 Performance Dojo
Investigate data center performance issues in style!

On the DVD

IPFire 2.29 Core Update 196

IPFire is an open source hardened Linux firewall distribution. Core update 196 introduces kernel and toolchain updates, console graphics stack modernization, and improved support for the WireGuard protocol. The kernel has been rebased to Linux 6.12.34, which brings improved hardware support, better performance, and various security enhancements. The text console was migrated to use the Linux Direct Rendering Manager (DRM) for a modern graphics stack. With the removal of bottlenecks in the kernel, IPFire can now handle each WireGuard device in an independent kernel thread so that all available CPU cores can be used to maximize throughput.

IPFire_
2.29
ISSUE 89/2025
ADMIN
Network & Security

News for Admins

Tech News

IBM and AMD Partner on Quantum-Centric Supercomputing Initiative

IBM and AMD are partnering to develop “computing architectures based on the combination of quantum computers and high-performance computing, known as quantum-centric supercomputing.”

“In a quantum-centric supercomputing architecture,” the announcement states (https://newsroom.ibm.com/2025-08-26-ibm-and-amd-join-forces-to-build-the-future-of-computing), “quantum computers work in tandem with powerful high-performance computing and AI infrastructure, which are typically supported by CPUs, GPUs, and other compute engines.”

Through this collaboration, the companies aim to develop “scalable, open source platforms” and to explore how open source ecosystems, such as Qiskit (https://www.ibm.com/quantum/qiskit), could catalyze development.

“Quantum computing will simulate the natural world and represent information in an entirely new way,” said Arvind Krishna, Chairman and CEO of IBM. “By exploring how quantum computers from IBM and the advanced high-performance compute technologies of AMD can work together, we will build a powerful hybrid model that pushes past the limits of traditional computing.”

Learn more at IBM (https://www.ibm.com/us-en).

Python Documentary Released

CultRepo has released a documentary relating the history of the popular Python programming language (https://www.python.org/psf-landing/).

The 90-minute documentary, which you can watch on YouTube (https://www.youtube.com/watch?v=GfH4QL4VqJ0), features accounts from Guido van Rossum, Travis Oliphant, Barry Warsaw, and many others. It tells “the story of Python’s rise, its community-driven evolution, the conflicts that almost tore it apart, and the language’s impact on … well … everything.”

The documentary notes that “what began as a side project in Amsterdam during the 1990s became the software powering artificial intelligence, data science, and some of the world’s biggest companies.”

Watch *Python: The Documentary | An origin story* (https://www.youtube.com/watch?v=GfH4QL4VqJ0) on YouTube.

NVIDIA Announces Spectrum-XGS Ethernet

NVIDIA has announced Spectrum-XGS Ethernet technology for connecting distributed data centers.

This addition to the NVIDIA Spectrum-X Ethernet platform (https://www.nvidia.com/en-us/networking/spectrumx/) introduces “scale-across infrastructure” aimed at extending the performance and scale of Spectrum-X Ethernet to “interconnect multiple, distributed data centers to form massive AI super-factories,” the announcement states (https://nvidianews.nvidia.com/news/nvidia-introduces-spectrum-xgs-ethernet-to-connect-distributed-data-centers-into-giga-scale-ai-super-factories).

Get the latest IT and HPC news in your inbox

Subscribe free to ADMIN Update
bit.ly/ADMIN-Update

Lead Image © vlastas, 123RF.com

Spectrum-XGS Ethernet is fully integrated into the Spectrum-X platform and features:

- Auto-adjusted distance congestion control
- Precision latency management
- End-to-end telemetry

"With NVIDIA Spectrum-XGS Ethernet, we add scale-across to scale-up and scale-out capabilities to link data centers across cities, nations, and continents," said Jensen Huang, founder and CEO of NVIDIA.

Learn more at NVIDIA: https://www.nvidia.com/en-us/networking/products/ethernet/.

Rackspace Adds AI Security Engine to Its Cyber Defense Center

Rackspace Technology has added the Rackspace AI Security Engine (RAISE) to its Rackspace Cyber Defense Center (RCDC) with the aim of accelerating threat detection and response.

According to the announcement (https://www.rackspace.com/newsroom/rackspace-unveils-raise-ai-driven-security-engine), RAISE combines "human expertise with machine-driven precision" to perform signal processing, detect patterns, and recommend remediation actions in response to perceived threats.

"This self-improving model reinforces effective strategies and refines future actions, enabling faster, smarter, and more resilient cybersecurity operations over time," the company says.

Learn more at Rackspace: https://www.rackspace.com/blog/rackspace-ai-security-engine-strengthens-cyber-defense-adaptive-intelligence.

Microsoft Contributes DocumentDB to the Linux Foundation

Microsoft has contributed DocumentDB – a PostgreSQL-based, open source document database – to the Linux Foundation, where it will be developed under the permissive MIT license.

Since its launch earlier in 2025, the flexible and extensible DocumentDB "has rapidly gained momentum within the developer community, accumulating nearly 2,000 GitHub stars and 100s of contributions, feedback, and adoption," the Linux Foundation said (https://www.linuxfoundation.org/press/linux-foundation-welcomes-documentdb-to-advance-open-developer-first-nosql-innovation).

"DocumentDB fills a critical gap in the document database ecosystem, attracting contributors, users, and champions. What's even more exciting is it provides an open standard for document-based applications, like what SQL did for relational databases," said Jim Zemlin, executive director of the Linux Foundation.

Learn more about DocumentDB on GitHub: https://github.com/documentdb/documentdb.

OCCTET Project Created to Simplify CRA Compliance

The Eclipse Foundation (http://www.eclipse.org/) has launched the OCCTET project with the aim of helping organizations handle compliance with the Cyber Resilience Act (CRA) (https://digital-strategy.ec.europa.eu/en/policies/cyber-resilience-act).

The Open Source Compliance: Comprehensive Techniques and Essential Tools (OCCTET) (https://occtet.eu/) project – funded by the European Commission – brings together open source advocates and developers "to build free, open source tools that make regulatory compliance more accessible, transparent, and cost-effective," the announcement says.

Namely, the OCCTET project intends to provide an open source toolkit along with other resources to help organizations automate compliance related to the use of open source software in digital products.

"Compliance with the CRA is a multi-year journey that organizations need to prioritize now," said Mike Milinkovich, executive director of the Eclipse Foundation.

Read more at the Eclipse Foundation: https://www.eclipse.org/.

Debian Announces Version 13 "Trixie"

The Debian project, which celebrated 32 years of development this month, has announced version 13 (code name Trixie) featuring:

- Official support for RISC-V (64-bit RISC-V64)
- Enhanced security on both AMD64 and ARM64

- HTTP Boot support in Debian Installer and Live images for UEFI/U-Boot systems
- Upgraded software stack: Gnome 48, KDE Plasma 6, Linux kernel 6.12 LTS, GCC 14.2, Python 3.13, and more

This version also ships with several desktop environments, including:

- Gnome 48
- KDE Plasma 6.3
- LXDE 13
- LXQt 2.1.0
- Xfce 4.20

Debian 13 will be supported for the next five years through the work of the Debian Security team (https://security-team.debian.org/) and the Debian Long Term Support (https://wiki.debian.org/LTS) team.

Learn more at Debian: https://bits.debian.org/.

CIQ Announces General Availability of Warewulf Pro

CIQ has announced the general availability of Warewulf Pro for building and managing compute clusters at scale.

Warewulf Pro (https://ciq.com/blog/introducing-warewulf-pro/) offers new features and customer support to “simplify and speed up the process of deploying and maintaining your HPC clusters,” including:

- Modern web-based interface built on Cockpit
- Curated, prebuilt node images
- Production-ready packages
- Full enterprise support for deployments, upgrades, and design

The new interface is designed to help cluster administrators quickly understand the underlying organization of a Warewulf cluster by showing:

- List of nodes to provision
- Configuration profiles that may be applied to those nodes
- Operating system images to provision to those nodes
- Configuration overlays that customize the provisioned images for each node
- In-line documentation for each entity and field

Learn more at CIQ: https://ciq.com/.

TuxCare Radar Vulnerability Scanner Released

TuxCare has released TuxCare Radar, “an AI-enhanced vulnerability scanner purpose-built for Linux-native environments.”

According to the company, TuxCare Radar (https://tuxcare.com/tuxcare-radar/):

- Can be rapidly deployed with a lightweight CLI
- Generates up-to-date scan reports
- Minimizes false positives
- Prioritizes real-world threats with AI-assisted risk scoring
- Offers remediation guidance

“Radar delivers accurate, real-time vulnerability detection and reporting that minimizes false positives by recognizing both traditional and rebootless patches in-memory,” the company says.

Learn more at TuxCare (https://tuxcare.com/).

CISA Releases Thorium for Automated Malware Analysis

The Cybersecurity and Infrastructure Security Agency (CISA), along with Sandia National Laboratories, has released Thorium (https://www.cisa.gov/resources-tools/resources/thorium) – “a scalable platform for automated file analysis and result aggregation.”

Thorium lets teams integrate commercial and open source analysis tools to assess malware threats and generate forensic analysis results. It is “configured to ingest over 10 million files per hour per permission group and schedule over 1,700 jobs per second, while maintaining a fast results query,” the announcement states (https://www.cisa.gov/news-events/news/cisa-announces-release-thorium-malware-analysis).

The platform lets you integrate command-line tools as Docker images and can be used for:

- Filtering tool results
- Controlling access through strict group-based permissions
- Defining event triggers and tool execution sequences to automate workflows
- Result aggregation
- Tool sharing

Learn more at Thorium: https://www.cisa.gov/resources-tools/resources/thorium.

Stack Overflow Survey: 66% of Developers Frustrated by AI Inaccuracy

Positive sentiment for AI has decreased among developers from more than 70 percent last year to 60 this year, says the 2025 Developer Survey (https://survey.stackoverflow.co/2025) from Stack Overflow.

This year's survey asked several questions about developers' adoption and use of AI and machine learning tools. Response highlights are shown below:

- Fifty-one percent of professional developers use AI tools daily.
- Eighty-four percent of respondents are using or are planning to use AI tools in their development process, up from 76 percent last year.
- Forty-six percent don't trust the accuracy of AI output, up from 31 percent last year.

The biggest frustration – cited by 66 percent of developers – is "AI solutions that are almost right, but not quite," followed by "debugging AI-generated code is more time-consuming" at 45 percent.

Overall, developers say they're using AI for tasks such as:

- Searching for answers (54.1%)
- Generating content or synthetic data (35.8%)
- Learning new concepts or technologies (33.1%)
- Documenting code (30.8%)
- Creating or maintaining documentation (24.8%)

The most frequently used LLM models are:

- OpenAI's GPT models (81%)
- Claude Sonnet models (43%)
- Gemini Flash models (35%)

For the 2025 survey, Stack Overflow received more than 49,000 responses from 177 countries, with most respondents citing more than 10 years of coding experience.

Years of coding experience (all respondents):

- 1-5 years (13.9%)
- 6-10 years (21.1%)
- 11-15 years (15.6%)
- 16-20 years (11.8%)
- 21-30 years (14.6%)
- 31-40 years (7%)
- 41-50 years (3.1%)
- More than 50 years (0.5%)

Read more at Stack Overflow: https://stackoverflow.co/.

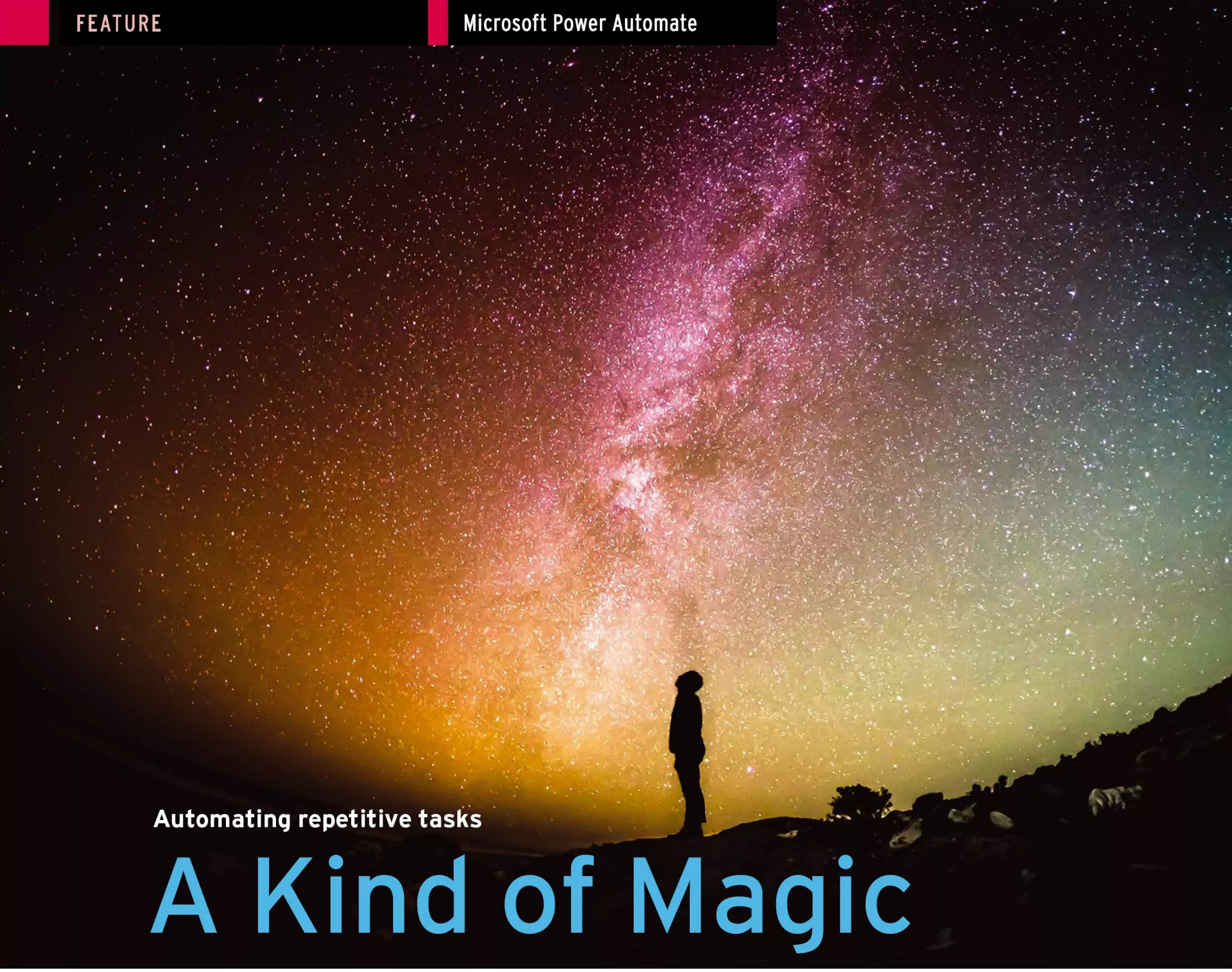
Intel Shuts Down Clear Linux OS

Intel is ending support for Clear Linux OS (https://www.clearlinux.org/), a performance-optimized, open source Linux distribution, according to a statement on the Clear Linux forum.

"After years of innovation and community collaboration, we're ending support for Clear Linux OS. Effective immediately, Intel will no longer provide security patches, updates, or maintenance for Clear Linux OS, and the Clear Linux OS GitHub repository will be archived in read-only mode," says Arjan van de Ven (https://community.clearlinux.org/t/all-good-things-come-to-an-end-shutting-down-clear-linux-os/10716) on the forum.

If you're using Clear Linux, van de Ven recommends "planning your migration to another actively maintained Linux distribution as soon as possible."

You can learn more about Clear Linux OS on the website: https://www.clearlinux.org/.



Automating repetitive tasks

A Kind of Magic

We provide an overview of the features, fields of application, and pricing of Microsoft Power Automate, highlighting the pros and cons of the SaaS platform for optimizing and automating workflows and business processes. By Philip Lorenz

Microsoft Power Automate (MSPA, previously Microsoft Flow) enables automated processes between apps and services, which greatly simplifies tasks such as synchronizing data, sending notifications, and capturing information. Processes can be optimized and recurring tasks automated without manual intervention by connecting different services.

Power Automate is intended for M365 admins, specialist departments, and business analysts, as well as users without in-depth programming knowledge. The user-friendly interface helps normal users independently create workflows to reduce the load on the IT department. Admins retain control, of course, and can enforce security policies and compliance standards.

MSPA is not a direct alternative to scripting languages (e.g., Bash, PowerShell, or Python) but focuses on integrating a variety of services and applications by means of connectors. MSPA is ideal for automating business processes that span multiple systems and platforms without the need to write code; instead, it acts as the glue between the applications where processes are designed in a modular way across the web user interface (UI) [1].

Power Platform Ecosystem

Power Automate is a central component of the Microsoft Power Platform and provides low-code tools. In combination with other tools, MSPA offers a comprehensive system that covers the entire process optimization cycle, from data collection and analysis, through process automation and the development of customized applications, to communication with external users. Power Automate is the automation engine that links the various components and enables an exchange of information and workflow automation.

Besides MSPA, the Power Platform includes Power Apps for creating custom applications without in-depth programming skills. Power BI provides data-driven insights and enables informed decisions on the basis of company data. Power Pages is a low-code platform for dynamic extranet websites that support external access. Finally, Power Virtual Agents enable interactive chatbots without any programming overhead; it is a great

Photo by Greg Rakozy on Unsplash

choice for automating customer and employee inquiries.

Set Up and Use

Power Automate comprises several components that are essential for creating flows:

- *Connectors* connect services and enable an exchange of data. Data gateways also facilitate the integration of on-premises applications. Users can even create their own connectors wherever they need to integrate custom services.
- *Triggers* start a flow when certain events occur. Triggers can be triggered on the fly, manually, on a schedule, or automatically.
- *Actions* are specific tasks that a flow handles, such as retrieving data, creating entries, or editing information.
- *Conditions* define the circumstances under which an action is executed, which means that workflows can be designed flexibly to fire up only in certain situations.
- *Templates* are flow templates that help you quickly implement frequently used automations.
- *Approvals* initiate workflows that allow decisions to be structured and documented.

MSPA is available in desktop and cloud versions, which pursue different automation goals and differ fundamentally in how they work and the options they offer. The cloud version is ideal when automation needs to be fast, flexible, and free from direct user interaction. It works on the basis of APIs and connects effortlessly to servers and software as a service (SaaS) services to execute automated processes.

Processes are triggered by specific events, such as an incoming email (Figure 1). For example, if you want to enter data from a new customer request automatically into an Excel spreadsheet or publish a status update in Microsoft Teams, cloud flows are the best choice. Thanks to numerous connectors for cloud and local services, these flows offer a wide range of potential applications.

Cloud flows are available in three categories. Automated flows are triggered by specific events and use cloud connectors to ensure that applications intercommunicate smoothly. In contrast, manual flows are launched at the push of a button (e.g., on a smartphone), which helps you automate recurring tasks such as sending reminders. Finally, scheduled flows take care of regularly recurring tasks. For example, you can schedule a daily data transfer to a database, and the flow ensures that this always happens at the specified time.

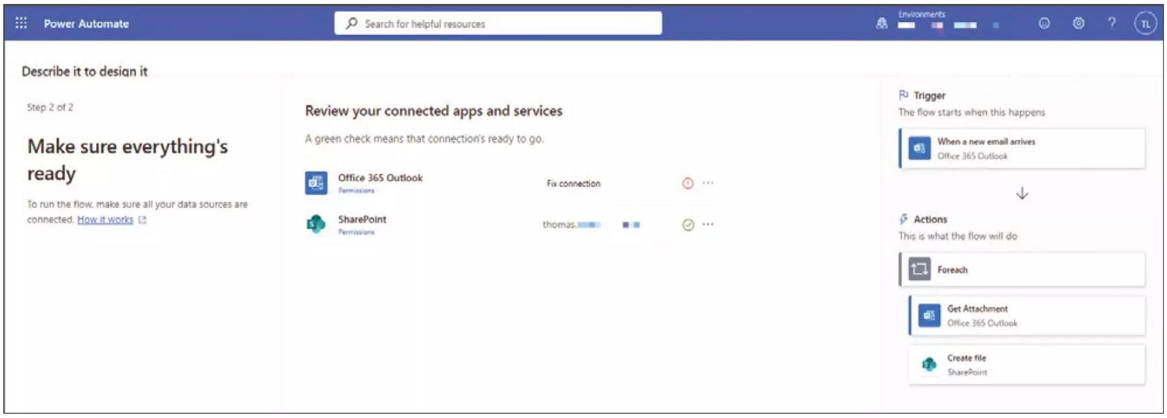
Desktop flows focus on robotics process automation (RPA) and aim to replicate human interactions on the computer. For example, desktop MSPA can learn exactly where to push the mouse button, what information to copy, or how to fill out certain forms. Desktop flows act like a human, making them ideal for processes that do not have APIs or require direct user interaction and are a great choice for applications that do not offer automation interfaces. MSPA can use visual elements of the desktop and even navigate in older applications that can only be controlled from a graphical user interface (GUI). The goal is to replicate human actions precisely and handle repetitive tasks efficiently.

Workflows in Practice

To bring the MSPA technology to life, I'll show you some practical examples, such as automatically saving email attachments to OneDrive. Whenever an email with an attachment arrives in a specific mailbox, it is automatically dropped into a defined folder in OneDrive. You can also be notified of new entries in an Excel spreadsheet. When a new entry is added to a specific Excel spreadsheet, Power Automate sends a notification to a Microsoft Teams group to inform the team.

Another option is to implement approval processes for vacation requests. When an employee submits a vacation request form, it is automatically sent to the approving supervisor, who is then notified by email or Teams. The supervisor can then approve or reject the request directly; the applicant is informed of the decision; and, if approved, the vacation is automatically added to a team calendar.

These examples showcase MSPA's greatest strength: the availability



Describe it to design it

Step 2 of 2

Make sure everything's ready

To run the flow, make sure all your data sources are connected. How it works

Review your connected apps and services

A green check means that connection's ready to go.

Office 365 Outlook — Permissions — Fix connection

SharePoint — Permissions — thomas.[redacted]

Trigger

The flow starts when this happens

When a new email arrives — Office 365 Outlook

Actions

This is what the flow will do

Foreach

Get Attachment — Office 365 Outlook

Create file — SharePoint

Figure 1: **Processes are created in Power Automate in an intuitive GUI and do not require any programming.**

of hundreds of connectors [2] that enable communication with various services and applications. In addition to Microsoft services such as SharePoint, Teams, and Outlook, external systems can be integrated, including ServiceNow, SAP, and Salesforce, as well as smaller services such as ToDoist and infrastructure-related components such as PostgreSQL and MongoDB. Power Automate also comes with numerous ready-to-use templates that help users get started quickly and automate recurring processes.

Workflows with AI

Microsoft is increasingly relying on artificial intelligence (AI) throughout the Power Platform and, logically, in MSPA as well. One example is Copilot Studio, an orchestrator for various AI-powered applications, such as provisioning chatbots (Figure 2). What is particularly exciting is that these chatbots can be combined with Power Automate to trigger specific actions directly.

AI also plays a crucial role in workflows. Users can simply describe in plain English what they want their workflow to look like. Copilot then creates a proposal for a Power Automate workflow and, assuming the user agrees with the proposal, the desired process is created. Moreover, the user is told all about details that still need to be added, such as the precise specification of a SharePoint page to be used as a resource.

Licensing

The Microsoft Power Automate license you choose will depend on your requirements and use cases. Microsoft offers two basic licensing models that cover these needs, whether for simple automations or complex, enterprise-wide processes. The licenses are dubbed Power Automate Premium and Power Automate Process. You will find a comprehensive overview of the M365 licensing universe and MSAP online [3].

Power Automate Premium is available for $15 per user and month. Organizations with this license can create and execute unlimited cloud flows for digital process automation (DPA). It also enables the use of desktop flows through robotic process automation in supervised mode, which is particularly useful when working with older applications.

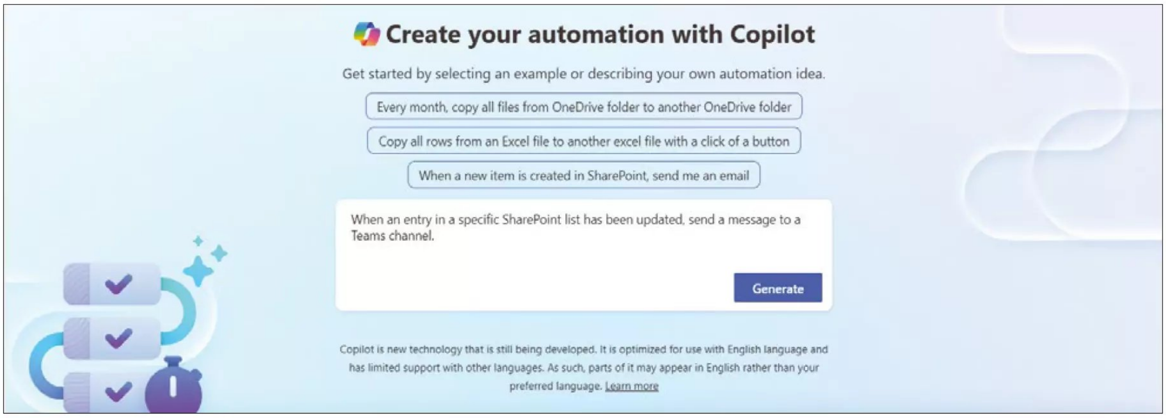
Additionally, the MSPA desktop license provides access to process mining on desktops and 50MB of process mining data storage, which can be expanded to up to 100GB per tenant. Power Automate Premium also provides AI tools that support tasks such as form processing, object recognition, predictions, and text recognition. However, you do need an additional MSPA Hosted Process license to run an RPA bot in unattended mode.

The Premium license is intended for users who want to create automations with premium or custom connectors and those who want to simulate applications that mimic human behavior (supervised RPA). Developers who create and manage desktop flows also require a Premium license.

One important aspect of Premium licensing is that automated and scheduled flows always run in the context of the flow owner, regardless of who starts the flow or which accounts are used for the connections. Instant flows (those triggered by a button press or trigger events) are executed in the context of the user who triggered the flow. If an automated flow uses premium connectors, only the owner needs to have a premium license, although it is different for instant flows: All triggering users require a premium license. A Process license can be a cost-effective alternative, because it eliminates the need to license individual users.

Power Automate Process costs $150 per bot and month and is designed for organizations that need autonomous bots for automated processes without manual intervention. This license is particularly suitable for centralized tasks that are available to all employees in the organization. A Process license covers all the flows within a business process, and the flows themselves can be managed in a coherent way. An MSPA Process license is required for:

- flows with unattended RPA to run desktop flows on machines without user interaction;
- workflows that run with an application account (service principal);



Create your automation with Copilot

Get started by selecting an example or describing your own automation idea.

Every month, copy all files from OneDrive folder to another OneDrive folder

Copy all rows from an Excel file to another excel file with a click of a button

When a new item is created in SharePoint, send me an email

When an entry in a specific SharePoint list has been updated, send a message to a Teams channel.

Generate

Copilot is new technology that is still being developed. It is optimized for use with English language and has limited support with other languages. As such, parts of it may appear in English rather than your preferred language. Learn more

Figure 2: Power Automate users can create workflows in combination with Copilot.

which is often the case in companies with multiple environments (e.g., development, test, and production) to transfer and manage flows between environments by DevOps pipelines without granting users direct access to the production environment;

- flows that process large volumes of data or start frequently, generating many Power Platform requests;
- large numbers of users who access a Premium flow, in which case, either all users need a Premium license or you cover this type of flow with a Process license, which offers organizations the option of aligning licensing with the number of flows (because of the fixed monthly price, this solution is particularly useful for larger organizations or MSPA implementations that affect many employees); and
- flows that use Premium connectors and are accessed by numerous guest users without it being clear whether they have Premium licenses in their home tenant.

Premium connectors can be used in both licensing models but differ in terms of the capacity available for database and file storage in Microsoft Dataverse (the storage environment for Copilot and MSPA). Power Automate Premium offers 250MB of database storage and 2GB of file storage, whereas the MSPA Process Plan includes 50MB of database storage and 200MB of file storage. These differences could play a role in choosing the right licensing model, depending on your organization's storage requirements.

Conclusions

Microsoft Power Automate is an impressive platform for automating work processes, especially for organizations that already rely on Microsoft 365. Thanks to its seamless integration with the Microsoft environment and its low-code approach, MSPA is a great choice for automating recurring tasks to help save time and resources. The wide range of connectors makes it possible to link internal Microsoft services seamlessly (e.g., Outlook, SharePoint, and Teams), as well as external applications, which opens a wide range of potential applications in various divisions of the organization. ■

Info

[1] Microsoft Power Automate docs: [https://learn.microsoft.com/en-us/power-automate/]

[2] All connectors: [https://learn.microsoft.com/en-us/connectors/connector-reference/connector-reference-powerautomate-connectors]

[3] M365 licenses: [https://m365maps.com/matrix.htm]

Shop the Shop shop.linuxnewmedia.com/shop

Missed an issue?

You're in luck.

Most back issues are still available. Order now before they're gone!

shop.linuxnewmedia.com/shop

GET IT NOW!

SAVE TIME ON DELIVERY WITH OUR ALTERNATIVE PDF EDITIONS

ADMIN
Network & Security
Non-Relational Databases
Monster VMs
Gitea
Zero Trust
grommunio
Storage
Purdue Model
Kubeflow
Monitoring
Data Obfuscation
Secure AD Design
Flipper Zero
Admin Distros
Local AI Server
A New Checkmk
5 Network Admin Distros
Lightweight Kubernetes
AD Risk Mitigation
IVRE

Red Hat Ansible Automation Platform

Stumbling Blocks

After acquiring Ansible, Red Hat integrated it into its portfolio and now offers the Ansible Automation Platform as a comprehensive automation and orchestration tool. We look at how the product performs in practice.

By Martin Loschwitz

Ansible is immensely popular among admins, not least because of its relatively easy-to-learn syntax. Anyone who has ever worked with Puppet or SaltStack knows the kind of infrastructure required to run these systems. Ansible takes a far more lightweight approach: All you need is SSH access to the command line – either directly as root or with `sudo` on the target system – to assume the admin role.

Ansible does not rely on the complicated declarative syntax that Puppet uses. Instead, Ansible instructions are more like commented shell scripts that list the tasks to be performed step by step in exactly the same order in which they will be executed on the system. Moreover, you have the option of adding an explanation for each step.

Enhanced AWX

If you just want to automate a few tasks on a target system, the feature set is fine. Things get more complicated when you need to integrate automation with an external ruleset, such as compliance and security requirements, and become difficult if you need granular rights management. If you want to allow different users to execute different commands on systems, the standard version of Ansible will not do the trick.

Ansible developers themselves identified this shortcoming long ago and introduced a solution to the problem in the form of AWX, an open source web-based user interface and task engine for managing Ansible automation. AWX wraps itself around Ansible on the command line and adds features. The new tricks include comprehensive role-based access control (RBAC) rights management, as well as the ability to create continuous integration and continuous deployment (CI/CD) pipelines for automated execution of specific tasks started fresh from clean environments. One result is that organizations can avoid the problem of some Ansible commands on non-standard systems having undesirable effects on the target hosts.

Although AWX is freely available as open source software, Red Hat also has a commercial product in its portfolio that comes with support: the Ansible Automation Platform (AAP; Table 1; Figure 1). AAP has seen a number of enhancements in the recent past. Whereas it used to be primarily a graphical user interface (GUI) for calling Ansible playbooks, many of the new features only found their way into the software in AAP 2.0.

What can you expect if you add AAP to your environment? What does AAP offer in terms of basic functionality compared with other products of a similar type, and what special features does it have on board? How does AAP simplify security and compliance? How well does it support admins in creating modern CI/CD environments? How flexible is AAP when it comes to connecting local or hyperscaler-based services? The

Table 1: Ansible Automation Platform

Objective	To create, execute, and manage automation processes
Manufacturer	Red Hat [www.redhat.com/en/technologies/management/ansible]
Price (one-year, est.)	Up to 100 managed systems: $14,000 (EUR11,500) with standard support; $20,000 (EUR15,500) with premium support
System requirements	Multiple systems for the various AAP components (virtualization possible), at least four CPUs per system (or physical CPU cores), 8GB of RAM, 20GB of disk space, and a supported Linux distribution or target platform (RHEL or OpenShift)

Lead Image © Dmitry Sunagatov, Fotolia.com

following analyses reveal what you can (and must) expect from AAP.

Scalable Basic Framework

Even at first glance, AAP clearly is far more than just a way to call up playbooks in Ansible format. In Ansible, a playbook is basically a kind of template that uses its own syntax to list a series of work steps for target systems and specify the parameters that apply to these calls. Individual functions do not need to be called directly but can be called by roles, which are separate entities in Ansible and comprise the configuration steps to be applied with the associated service configuration and any additional operations.

AAP goes far beyond this description. When you install the platform, you are adding a variety of additional services to your environment that AAP needs to be able to run. One core component in the overall AAP construct is Redis, which acts as a consensus algorithm in the background to make sure that all systems belonging to AAP always have the same information about all managed systems and resources. A central aspect of the basic package is that the entire platform scales seamlessly across the board. You can roll out instances of the Platform Gateway, which controls all access to AAP, matching the automation logic in the Automation hub container. The event-driven Ansible component, which executes Ansible commands on target systems on demand, also scales seamlessly. Conveniently, after rolling out AAP, you only need to worry about some of these details. As long as you provide enough systems for AAP, it takes care of its own scalability in a largely independent way to make sure that the user experience is as good as possible.

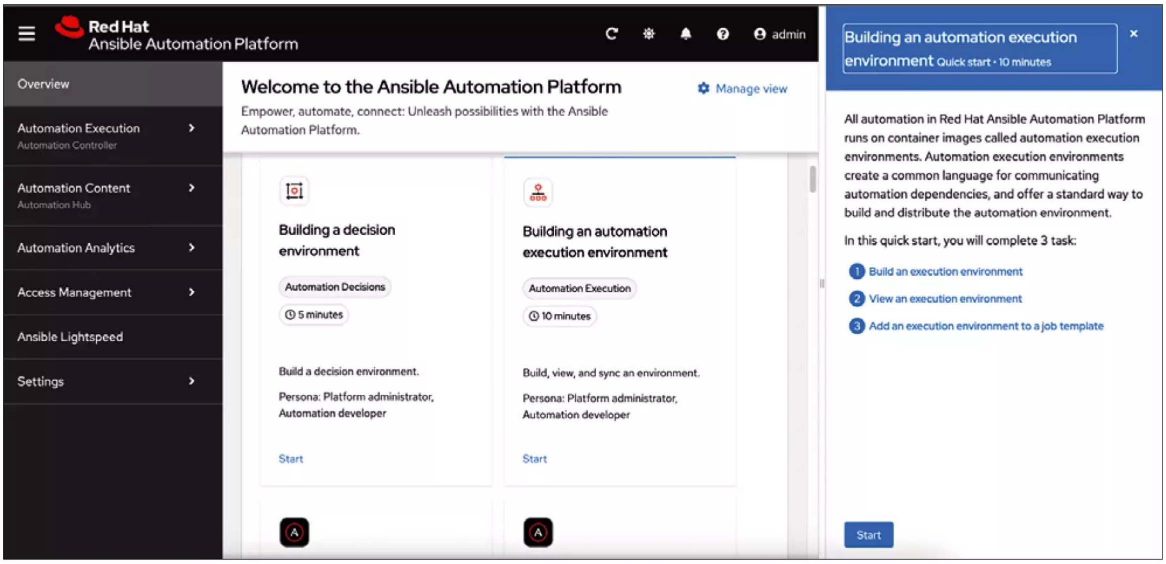
Event-Driven Ansible

Users normally access AAP from the web GUI, which is very similar to other tools offered by Red Hat for today's IT landscape, because the structure and logic are largely the same as those found in other products. Anyone familiar with the OpenShift GUI, for example, will be able to find their way around AAP easily. The list of menu items on the left provides quick access to the key functions of AAP, including Event-Driven Ansible (EDA), which automates the execution of Ansible calls, and the Automation Hub, a kind of marketplace for Ansible integrations curated by the manufacturer.

Core functionality is contained in the Automation controller and EDA in particular. Playbooks can be stored in the style of a CI/CD pipeline and then assigned to systems acquired and managed by AAP, configured to suit your needs, and finally executed. Usefully, AAP can manage various target platforms, each with their own inventories, and assign to them specific Ansible configurations, which is a massive help when it comes to keeping track of larger environments and implementing an organizational solution that you only need to define once. All told, the AAP scope that comprises the largest number of Automation controller features comes very close to a complete CI/CD environment with Ansible, because external GitHub directories can be seamlessly integrated at this point to implement infrastructure-as-code systems.

EDA is now also part of the AAP core. The software has not been around quite as long as Ansible itself, but it already enjoys huge popularity because EDA allows the systems of an Ansible inventory to be monitored specifically for individual events. You define the rules (Figure 2), and if a specific event occurs or the systems managed by EDA enter a state defined previously by a ruleset, AAP automatically triggers an Ansible run to restore the desired state.

All told, the Ansible Automation Platform's basic functionality is pretty


Red Hat
Ansible Automation Platform
admin
Overview
Automation Execution
Automation Controller
Automation Content
Automation Hub
Automation Analytics
Access Management
Ansible Lightspeed
Settings
Welcome to the Ansible Automation Platform
Manage view
Building a decision environment
Automation Decisions
5 minutes
Building an automation execution environment
Automation Execution
10 minutes
Start
Start
Building an automation execution environment Quick start • 10 minutes
Start

Empower, automate, connect: Unleash possibilities with the Ansible Automation Platform.

Build a decision environment.

Persona: Platform administrator, Automation developer

Build, view, and sync an environment.

Persona: Platform administrator, Automation developer

All automation in Red Hat Ansible Automation Platform runs on container images called automation execution environments. Automation execution environments create a common language for communicating automation dependencies, and offer a standard way to build and distribute the automation environment.

In this quick start, you will complete 3 task:

1. Build an execution environment
2. View an execution environment
3. Add an execution environment to a job template

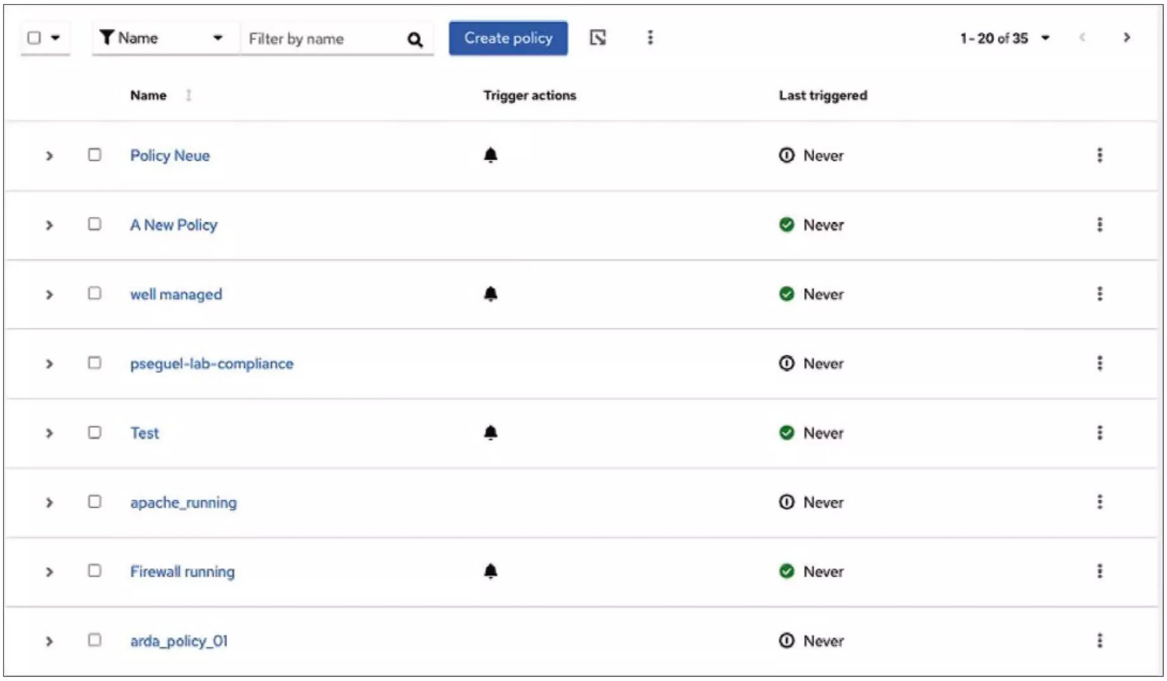
Figure 1: The web GUI is the heart of the Ansible Automation Platform and offers a range of support features.

much what you would expect from an enterprise product from a vendor such as Red Hat. Thanks to EDA, entire fleets of systems can be meaningfully automated in terms of software and configuration and kept in the desired states permanently.

Not everything is perfect on the AAP front, though. If you primarily use Ansible at the command line and encounter AAP for the first time, you will quickly notice that, although you will largely see the same terms and it is clear that familiar tools such as `ansible-playbook` are running somewhere under the hood, the GUI, which is the preferred mode of use for AAP, makes many things significantly more complicated than they would be if you were to use the shell, which is, of course, also because AAP is capable of doing far more.

Complex CI/CD Features

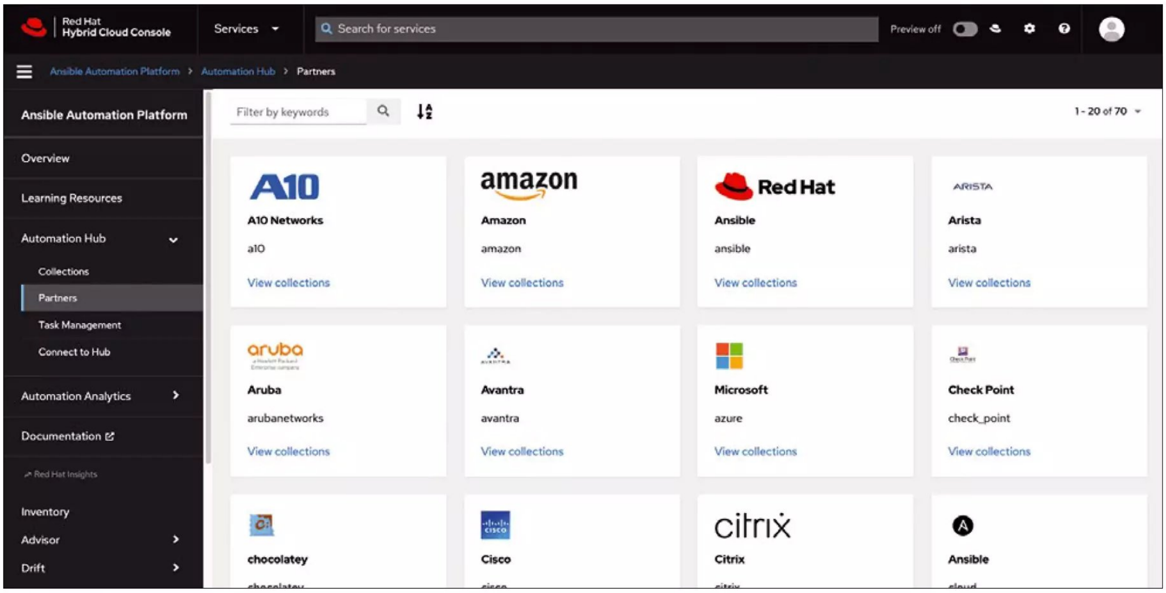
Various additional onboard AAP functions definitely go beyond the scope of typical Ansible functions, starting with the components that turn Ansible into a kind of CI/CD platform.



☐ ▾
Name ▾
Filter by name
Create policy
1 - 20 of 35 ▾

		Name	Trigger actions	Last triggered	
›	☐	Policy Neue	🔔	ⓘ Never	⋮
›	☐	A New Policy		✓ Never	⋮
›	☐	well managed	🔔	✓ Never	⋮
›	☐	pseguel-lab-compliance		ⓘ Never	⋮
›	☐	Test	🔔	✓ Never	⋮
›	☐	apache_running		ⓘ Never	⋮
›	☐	Firewall running	🔔	✓ Never	⋮
›	☐	arda_policy_01		ⓘ Never	⋮

Figure 2: The Insights feature gives you a comprehensive view of your own automation setup, for which you define the rules.



Red Hat
Hybrid Cloud Console
Services ▾
Search for services
Preview off
Ansible Automation Platform › Automation Hub › Partners
Ansible Automation Platform
Overview
Learning Resources
Automation Hub
Collections
Partners
Task Management
Connect to Hub
Automation Analytics
Documentation
Red Hat Insights
Inventory
Advisor
Drift
Filter by keywords
1 - 20 of 70 ▾

A10 Networks
a10
View collections

Amazon
amazon
View collections

Ansible
ansible
View collections

Arista
arista
View collections

Aruba
arubanetworks
View collections

Avantra
avantra
View collections

Microsoft
azure
View collections

Check Point
check_point
View collections

chocolatey

Cisco

Citrix

Ansible

Figure 3: The Automation Hub, a kind of marketplace, displays directly installable playbooks and roles curated by the manufacturer.

Admittedly, CI/CD is rarely a source of real pleasure for admins. Anyone who has ever had to manage Jenkins or Argo CD will be all too familiar with the issues.

That said, the tools are not usually to blame. The classic pipeline for CI/CD setups triggers an automated process by checking work into a Git directory, for example, and processing various work steps one after the other. Ideally you will get a working, nicely configured production instance of a system or a specific service, but the whole thing is inherently hugely complex.

AAP steps in here to offer admins various helpers, such as built-in version management for the repositories and the stored pipelines themselves, which also makes rolling back easier if a change to a pipeline goes awry. You can also start individual parts of a CI/CD pipeline independently from AAP in a web GUI, which is helpful if you are dealing with a large pipeline in which an error has crept in toward the end. This way, AAP avoids having to start a seemingly endless pipeline all over again.

However, handling pipelines, their versioning, and everything related to them has some drawbacks. If you set up AAP for this purpose, you first have to do some homework to understand the central strategies and then apply them in the GUI. A wizard would be useful in some places to help with the initial setup or with pipelines.

All told, AAP offers comprehensive options for automatically feeding software and the associated configurations to systems through CI/CD pipelines. The goal is also to realize a functioning system that meets your requirements.

Well-Curated Marketplace

Experienced Ansible users are familiar with the problem of turning to Google when looking for the right Ansible role or a suitable module for a specific task. The first few hits usually yield something useful, but the initial euphoria tends to be quickly followed by disillusionment: Ansible roles that someone posted online years ago and has not updated since were either perfect from the start, or their original author has since lost interest and the code is gathering dust.

Red Hat is trying to tackle the problem with Automation Hub (Figure 3), which collects integrations for a wide range of applications directly in Ansible. In practical terms, it is a kind of Ansible universe that is exclusively available to and curated for the AAP clientele. What you find in the Automation Hub is far more likely to work and be actively maintained than some random Google match. Moreover, the Automation Hub is well integrated into AAP, so you just need a few mouse clicks to add the specific code.

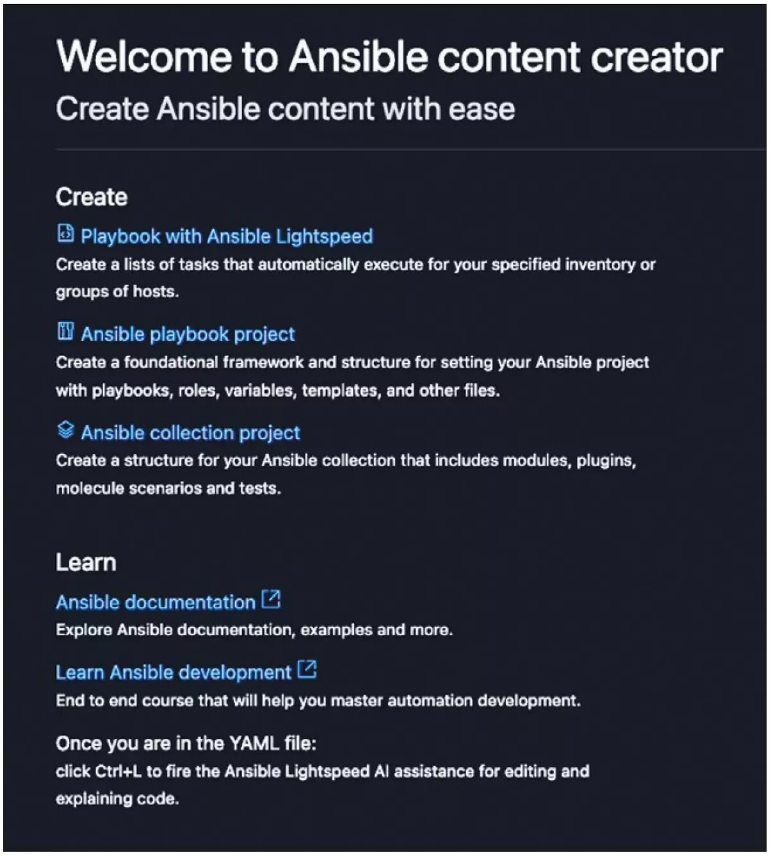
In practical terms, Red Hat is making an efficient contribution toward improving operational safety, security, and sustainability for its Ansible customers with AAP.

Fast AI and Successful Analysis

No article on automation in 2025 would be complete without mentioning artificial intelligence (AI).

Of course, Red Hat is in the thick of things when it comes to AI: Ansible Lightspeed is backed by IBM watsonx and is designed to help you write Ansible code more quickly and without external help (Figure 4). Lightspeed continuously monitors the environment and, in simple terms, guesses what you might want to do next.

These guesses might seem spooky at first, especially for people who are a little skeptical of AI and have not yet investigated the topic in any great detail. In a few quick tests, though, Lightspeed managed to predict useful Ansible code and complete the task at hand quickly. Of course, the technology is still in its infancy, but what it can already do today is very impressive, especially because no other



Welcome to Ansible content creator

Create Ansible content with ease

Create

Playbook with Ansible Lightspeed

Create a lists of tasks that automatically execute for your specified inventory or groups of hosts.

Ansible playbook project

Create a foundational framework and structure for setting your Ansible project with playbooks, roles, variables, templates, and other files.

Ansible collection project

Create a structure for your Ansible collection that includes modules, plugins, molecule scenarios and tests.

Learn

Ansible documentation

Explore Ansible documentation, examples and more.

Learn Ansible development

End to end course that will help you master automation development.

Once you are in the YAML file:

click Ctrl+L to fire the Ansible Lightspeed AI assistance for editing and explaining code.

Figure 4: Modern AAP versions integrate IBM watsonx in the background, offering the option of having Ansible code generated by AI.

software currently on the market offers comparable functionality.

Ultimately, it is important to Red Hat that an AAP admin always knows what is going on in their environment, so Automation Analytics, a cloud service that is part of AAP, regularly scans the environment and collects various critical values. It is linked to EDA in the background, which issues any event notifications; on the basis of these notifications, EDA launches. Automation Insights is also connected to the mother ship, Red Hat Insights. The software aims to make existing problems more visible to admins, proactively prevent issues through compliance requirements, and even paint colorful graphs that show you how much money all this fun is saving on the expense side.

Anyone who has used Red Hat Insights before will probably find Automation Insights useful, but you should think of it as a convenient add-on, although a decision for or against AAP is unlikely to depend on this feature.

Automatic Security and Compliance

By its very nature, AAP exists to enable compliance and regulated processes that are difficult or impossible to achieve by relying on Ansible's built-in tools alone. The provider again offers useful support. For example, if you want to connect AAP to an existing user directory, you have a wide range of options, including connecting to LDAP or Active Directory. AAP itself is a highly standardized product.

If set up correctly, the installation comes with various guarantees and commitments, which can ideally be passed on to your auditor and be used to implement and ultimately enforce central compliance requirements. Ansible through EDA is encroaching on the territory of classic compliance enforcers such as Chef's InSpec. Because AAP also implicitly offers the option of building CI/CD pipelines, it can be a central building block in an enterprise-wide immutable infrastructure agenda.

EDA and Automation Insights also can be used to identify and flag various security issues on your platform or eliminate them automatically. As a result, AAP makes a genuine contribution toward the overall security of an organization's systems. AAP comes with a component that automatically detects outdated versions of installed software fraught with vulnerabilities and automatically triggers a defined incident response process.

If you want to take things to the next level in a Red Hat environment, you can combine AAP with components such as Satellite, for managing and provisioning RHEL environments, to achieve an even higher degree of automation. Last but not least, AAP can be used as part of a security information and event management (SIEM) concept. With the use of its built-in feature for aggregating logfiles from intrusion detection (IDS) or firewall systems, for example, it acquires and evaluates relevant information and acts on the basis of defined automation. Of course, it does not guarantee sure-fire success, and it will take you some time to plan and implement this kind of setup.

Smooth Interaction with External Services

The question ultimately arises as to how well AAP integrates with external services considering two dimensions: On one hand, you could source AAP as a hosted service from a hyperscaler or run it as an application on an orchestration platform, which Red Hat is obviously aware would make the setup far easier. For example, they have comprehensive instructions on how to roll out and operate AAP on OpenShift, Red Hat's Kubernetes distribution. Those who purchase both AAP and OpenShift from Red Hat will naturally pay subscription fees to the provider for both products – it's just a kind of internal portfolio upsell.

On the other hand, you can purchase AAP as a hosted service with AWS or Azure, which also saves some headaches at setup; however, in return, you will have ongoing running costs that you need to factor into your budget planning to calculate a realistic bottom line. Although AAP supports manual deployment, the provider advises against this path.

Another dimension of interaction with other orchestrators is the ability to initiate actions from within AAP that in turn generate specific results on target platforms such as AWS or Kubernetes. AAP takes advantage of the comprehensive Ansible integrations for interaction with both Kubernetes and the major hyperscalers that can create, manage, or remove resources. The effect is massive. In fact, you can centrally store access credentials for a large number of target platforms, including VMware vCenter and OpenStack, meaning the platform in question can be defined as a deployment target as your next step (e.g., as part of a pipeline). Again, the integration is largely seamless and well implemented.

Conclusions

The Ansible Automation Platform is a safe bet for organizations looking to do more with Ansible than just run a few Ansible playbooks, especially because it has virtually no competition. In terms of its nature and scope, AAP is undoubtedly the most comprehensive product of its kind. However, it does not use this as an excuse to do just the bare minimum: If you need a management layer to wrap around Ansible, are looking to build pipelines, or simply need greater insights into the status of your automation, AAP offers useful features. The GUI could be improved because it is not very intuitive in places and requires some training; however, once you get the hang of it, AAP quickly delivers usable and measurable results. ■

The Author

Martin Loschwitz is the founder and managing director of True West IT Services GmbH, which offers scalable IT infrastructure based on OpenStack and Kubernetes.

Support for Windows 10 ends on 14 October 2025

Hundreds of millions of functioning computers may become e-waste, while new computers are produced unnecessarily. Production of new devices accounts for up to 75% of greenhouse gas emissions over their lifespan.

your computer
new software!

The most sustainable computer is the one you already own

Save your computer!

Make your computer faster and save it from the landfill with efficient, independent, Free & Open Source Software.

https://endof10.org

Published by Opt Green, a project from KDE Eco funded by UBA and BMUKN.

Aggregate alerts from various monitoring systems

Sound the Alarm

The free Alerta fields alerts from several monitoring systems and delivers them in a web GUI or at the command line. By Markus Stubbig

Historically grown environments often use more than one monitoring system. For example, production might use Zabbix, while research uses Ganglia with its computing clusters, and IT swears by Observium. In other cases, monitoring runs locally on each of the organization's sites and, of course, the tools are from different vendors because the various sites purchased them independently.

The Alerta [1] alerting manager collects messages from all of these systems, summarizes and de-duplicates them, and displays the results on a clear-cut web page. The Alerta web view gives you the messages from all the monitoring applications involved, sorting the messages by criticality. If you don't want to commit right away, the demo page [2] will give you a taste of the features.

Installing Alerta

Alerta is based on the Python programming language with Flask as its web framework. The data can be stored in either MongoDB or PostgreSQL. For actions on the command line, a command of the same name can generate alerts. You can save yourself some work with the `docker-alerta` Docker image [3] from the GitHub repository.

To begin, open the `docker-compose.yml` file and change the value for `ADMIN_PASSWORD`; then, start the containers:

```
wget https://raw.githubusercontent.com/➥
  alerta/docker-alerta/refs/heads/master/➥
  docker-compose.yml
sed -i ➥
    -e 's/ADMIN_PASSWORD=.*/➥
        ADMIN_PASSWORD=abc123/' ➥
    docker-compose.yml
docker-compose up -d
alias alerta="docker exec ➥
  -it root_web_1 /venv/bin/alerta"
alerta version
```

The final command outputs the version number and automatically checks whether the containers are running and whether internal communication in Alerta is working.

Now access the web interface on *http://<Your_Server>:8080/* and log in to the website with the access credentials from the Compose file. The alarm list that is now visible is initially empty. All that is missing are a few test alerts to bring the website to life. In principle, any monitoring system can be integrated by the Application Programming Interface (API); however, I will start with the `alerta` command.

Alarms at the Command Line

Alerta comes with a command of the same name, which not only sends alerts but is a full-fledged replacement for the web console – albeit in the console view. The command is part of the Docker installation but can also be used separately with the `pip` Python installer [4]. The Alerta server and the `alerta` command use the same API key by default, so you can set off the first alarm:

```
alerta send ➥
  --resource <Your_Server> ➥
  --service Disk ➥
  --event DiskFull ➥
  --environment Production ➥
  --severity major
```

If the two components are running on different hosts or use different API keys, you will need to adjust the local configuration. The `alerta` command takes its parameters from the `alerta.conf` file in your home directory or from environment variables. Both options are described and illustrated with examples in the GitHub repository. The

```
alerta top
```

command provides a quick overview of critical alarms (Figure 1). Unfortunately, the view cannot be filtered as

Lead Image © Chittima Kasa, 123RF.com

nicely as with the web console, which I introduce in the next section.

Color Highlighting in the GUI

The Alerta graphical user interface (GUI) appears without encryption in the web browser on *http://<Your_Server>:8080/*. After logging in, the test alarm of the `alerta` command should be visible. The default filter only shows open messages (i.e., messages for which the client has not yet given the all-clear). On the right-hand side of the respective alarm line, the message can be conveniently confirmed or deleted with the mouse.
Alerta pulls out all the stops when it comes to color highlighting: Alerts with a Critical weighting are red, Warnings are blue, Major is orange, Minor is yellow, and Normal is green. An alarm remains in the list until the client sends another message with a Normal severity level. The icon at top left opens the menu structure, where you can create users, groups, and API keys. Here, you can also generate an API key to integrate your own monitoring system.

Integrating Monitoring Systems

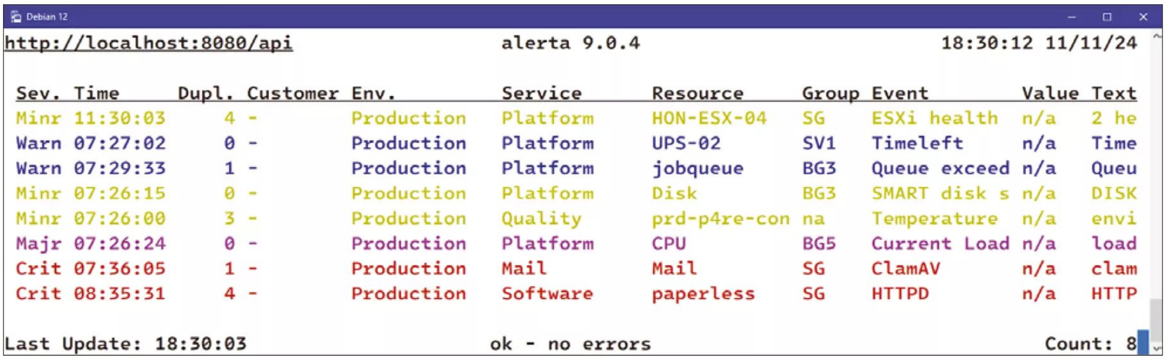
Nagios might not be the newest kid on the monitoring block, but it is certainly the best known. The Nagios-to-Alerta gateway **[5]** on GitHub is an event broker that sends Nagios alarms to Alerta with minimal overhead. An existing Nagios environment requires the precompiled `alerta-neb.o.nagios4` file on the local filesystem. You need to include this as a `broker_module` in your `nagios.conf` file and add the API key and the URL of the Alerta server in the same line. After restarting Nagios, a copy of each message is sent to Alerta. The event broker takes care of authentication and the correct format.
If you cannot find a ready-made integration project for your monitoring software, you will need to program it yourself – which sounds more complicated than it is, as long as the monitoring system can trigger an arbitrary command as an alert. This command then calls the `alerta` command and sends the alert to the Alerta server. For example, the M/Monit monitoring tool on a local Rocky Linux checks the HTTPD process and contacts Alerta if the web service becomes unavailable. The M/Monit-style configuration is:

```
check process httpd with pidfile ➘
  /run/httpd/httpd.pid
  if 2 restart within 3 cycle then ➘
  exec "/usr/local/bin/alerta ➘
        --config-file /etc/alerta.conf ➘
        --profile production send ➘
        --resource Rocky9 ➘
        --service Web ➘
        --event ServiceDown ➘
        --environment Production ➘
        --severity major"
```

Unfortunately, M/Monit does not give the all-clear, but at least it tries to restart several times before Alerta is notified of the failure.
Nagios and M/Monit are just two examples of how Alerta can be used in combination with other tools. The documentation **[6]** has a long list of compatible providers and software projects under *Integrations & Plugins* and in the GitHub repository, with candidates including Grafana, Telegram, Slack, Zabbix, and Fail2ban. Extensions such as *pinger* and *urlmon* let Alerta independently check the remote station for availability and generate alerts from the results. When browsing the list, the direction is always important, because Alerta can only receive SNMP traps and syslog messages, but cannot send them – with email and Microsoft Teams, it is exactly the opposite.

Manage Users

The logged-in administrator can do and see everything. A new user is created in the web GUI in the Users area and is assigned permissions with the selected role *user* or *guest*. What sounds like a fairly rough and ready authorization solution can be made more granular in the Permissions section. Roles are assigned specific rights for individual website functions, such as read access for alerts or modify rights for API keys.
By default, users can create their own accounts on the website and then work with the GUI. Self-registration can be disabled but only in the Compose file, where you need to set the `SIGNUP_ENABLED` variable to `False` and restart the containers. The registration option then disappears from the login page.
Because local user accounts are a nuisance in large environments, Alerta allows users to log in to an authentication server. The options



```
http://localhost:8080/api                alerta 9.0.4                 18:30:12 11/11/24

 Sev. Time      Dupl. Customer Env.         Service     Resource     Group Event        Value Text
 Minr 11:30:03     4 -         Production   Platform    HON-ESX-04   SG    ESXi health  n/a   2 he
 Warn 07:27:02     0 -         Production   Platform    UPS-02       SV1   Timeleft     n/a   Time
 Warn 07:29:33     1 -         Production   Platform    jobqueue     BG3   Queue exceed n/a   Queu
 Minr 07:26:15     0 -         Production   Platform    Disk         BG3   SMART disk s n/a   DISK
 Minr 07:26:00     3 -         Production   Quality     prd-p4re-con na    Temperature  n/a   envi
 Majr 07:26:24     0 -         Production   Platform    CPU          BG5   Current Load n/a   load
 Crit 07:36:05     1 -         Production   Mail        Mail         SG    ClamAV       n/a   clam
 Crit 08:35:31     4 -         Production   Software    paperless    SG    HTTPD        n/a   HTTP

Last Update: 18:30:03                  ok - no errors                                 Count: 8
```

Figure 1: Alerta displays the `alerta top` alarm list on the console or over SSH.

range from OpenID Connect, to SAML and OAuth2, to classic LDAP, which means you can use the most common authentication services or your local Active Directory. The configuration is handled at the command line in the `alertad.conf` configuration file. The `DEBUG` variable supports troubleshooting by enriching the output of

```
docker-compose logs -f
```

with detailed log messages.

Setting Up Multiclient Capability

Alerta can assign incoming alerts to clients. The corresponding user account only sees the messages from its own monitoring system in the web view, which is useful for managing multitenant setups. The assignment is made by the API key. If Alerta receives an alert that has been authenticated with the API key belonging to customer A, the message obviously belongs to that tenant. This distinction is also useful when multiple monitoring systems in the same company report to Alerta. In this case, each individual system is a tenant, so the web view always shows the environment in which the alert originated.

Alerta only uses this multitenant capability if the `CUSTOMER_VIEWS` variable is set to `True` in the Compose file. To enable, you need to restart the container. The web interface will then display the new Customers section. Create a Customer for each customer or monitoring system; then, create a new API key for each entry. Figure 2 shows how Alerta collects messages from multiple systems and displays the origin in the Customer column.

Alerta in the Cloud

To log the monitoring systems of different customers centrally, you can host the Alerta server with a cloud provider and receive the alerts from the Internet. However, you have to do some upgrading of the existing environment, because the server is currently unencrypted. You can use an upstream web server such as Caddy to resolve the issue. The Caddy software automatically obtains TLS certificates from Let's Encrypt to prevent warnings in the web browser. The configuration takes just a few lines. However, first you need to install Caddy on your local Linux machine with either the package manager, an executable, or a container. The web server expects its configuration to be stored in a text file named `Caddyfile`. Caddy only needs a few lines to help it perform its role as a reverse proxy:

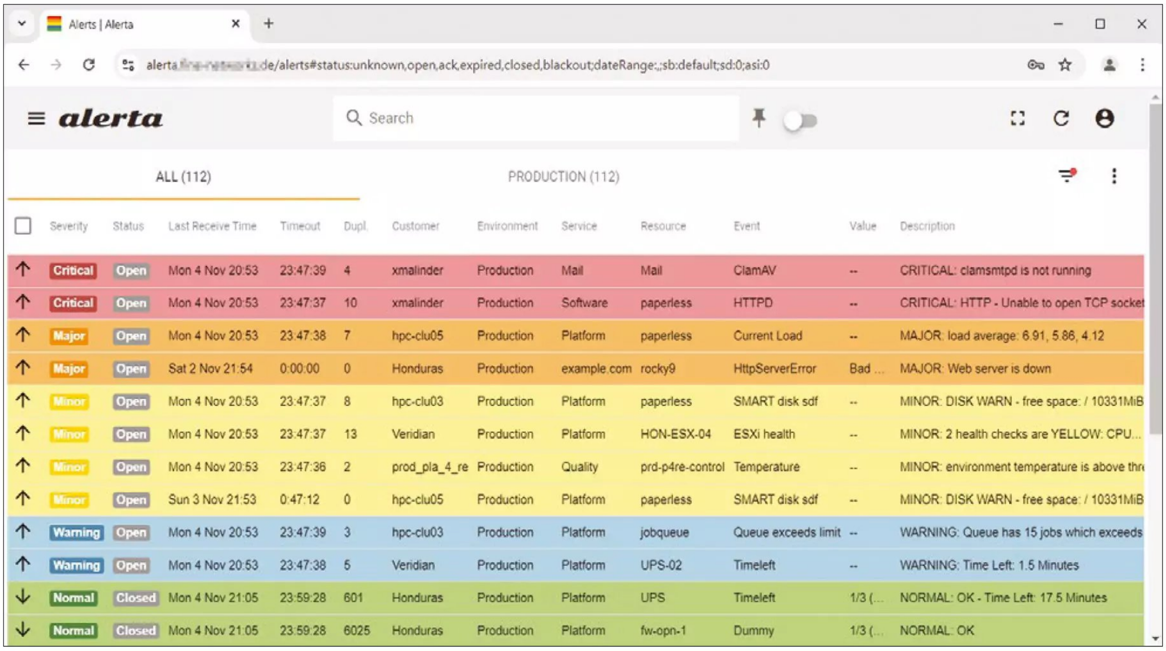
```
https://<alerta.example.net> {
    reverse_proxy localhost:8080
    # tls internal
}
```

Replace the first line with your server's DNS name. When you call

```
caddy run
```

the first time, Caddy contacts the Let's Encrypt certificate server and obtains valid TLS certificates for the name selected in the first line. If you do not have a public domain or are trying out the setup on your local network, remove the hash in line 3. Caddy will then use self-signed certificates. Alerta is then accessible on *https://<alerta.example.net>* – both the website and the API. If the hosting provider has a firewall, it needs permission for ports 80 (Let's Encrypt) and 443 (HTTPS).

Caddy can also be started as a Docker container and integrated into the


Alerts | Alerta
alerta.[redacted].de/alerts#status:unknown,open,ack,expired,closed,blackout;dateRange:,;sb:default;sd:0;asi:0
alerta
Search
ALL (112)
PRODUCTION (112)

☐	Severity	Status	Last Receive Time	Timeout	Dupl.	Customer	Environment	Service	Resource	Event	Value	Description
↑	Critical	Open	Mon 4 Nov 20:53	23:47:39	4	xmalinder	Production	Mail	Mail	ClamAV	--	CRITICAL: clamsmtpd is not running
↑	Critical	Open	Mon 4 Nov 20:53	23:47:37	10	xmalinder	Production	Software	paperless	HTTPD	--	CRITICAL: HTTP - Unable to open TCP socket
↑	Major	Open	Mon 4 Nov 20:53	23:47:38	7	hpc-clu05	Production	Platform	paperless	Current Load	--	MAJOR: load average: 6.91, 5.86, 4.12
↑	Major	Open	Sat 2 Nov 21:54	0:00:00	0	Honduras	Production	example.com	rocky9	HttpServerError	Bad ...	MAJOR: Web server is down
↑	Minor	Open	Mon 4 Nov 20:53	23:47:37	8	hpc-clu03	Production	Platform	paperless	SMART disk sdf	--	MINOR: DISK WARN - free space: / 10331MiB
↑	Minor	Open	Mon 4 Nov 20:53	23:47:37	13	Veridian	Production	Platform	HON-ESX-04	ESXi health	--	MINOR: 2 health checks are YELLOW: CPU...
↑	Minor	Open	Mon 4 Nov 20:53	23:47:36	2	prod_pla_4_re	Production	Quality	prd-p4re-control	Temperature	--	MINOR: environment temperature is above thr
↑	Minor	Open	Sun 3 Nov 21:53	0:47:12	0	hpc-clu05	Production	Platform	paperless	SMART disk sdf	--	MINOR: DISK WARN - free space: / 10331MiB
↑	Warning	Open	Mon 4 Nov 20:53	23:47:39	3	hpc-clu03	Production	Platform	jobqueue	Queue exceeds limit	--	WARNING: Queue has 15 jobs which exceeds
↑	Warning	Open	Mon 4 Nov 20:53	23:47:38	5	Veridian	Production	Platform	UPS-02	Timeleft	--	WARNING: Time Left: 1.5 Minutes
↓	Normal	Closed	Mon 4 Nov 21:05	23:59:28	601	Honduras	Production	Platform	UPS	Timeleft	1/3 (...	NORMAL: OK - Time Left: 17.5 Minutes
↓	Normal	Closed	Mon 4 Nov 21:05	23:59:28	6025	Honduras	Production	Platform	fw-opn-1	Dummy	1/3 (...	NORMAL: OK

Figure 2: Alerta distinguishes between clients for incoming alerts, allowing monitoring systems within an organization to be identified.

existing Compose file. The extended `docker-compose.yml` file is available for download **[7]**.

Targeted API Use

With Alerta, everything relies on the API, even if this is not apparent at first glance. The web front end communicates with the back end on localhost and transfers all the commands to the API in JSON format. Although not a groundbreaking innovation, it demonstrates the developers' structured approach. API access is open but requires authentication in the form of an API key. This type of key is the same as that used by a monitoring system to transfer alarms to Alerta. In the simplest case, `curl` accesses the API and retrieves an overview of the current situation (**Listing 1**).

The command chain with `jq` makes the output a bit friendlier; otherwise everything would be in a single long line. Also interesting is the alert endpoint, which the API uses to receive an alert without the `alerta` command having to be installed on the local machine. In this way, Alerta can be operated entirely with `curl` or integrated into your software.

The official documentation explains all the functions and provides detailed examples in the *API Reference* section. If you want to practice safely in a demo environment, take a look at the API kit **[8]**. If you encounter any initial difficulties, a request to the endpoint */management/healthcheck* will check whether the Alerta service is running and the database is OK. Ideally, you will receive a *200 OK* response.

Listing 1: Retrieving Data

```
curl --header 'Authorization: Key 1aSTt<...>'
   https://<alerta.example.net>/api/alerts/count | jq
{
    "autoRefresh": true,
    "severityCounts": {
        "critical": 2,
        "major": 1,
        "minor": 4,
        "normal": 101,
        "unknown": 1,
        "warning": 2
    },
    "status": "ok",
    "statusCounts": {
        "closed": 101,
        "open": 10
    },
    "total": 111
}
```

Checking Whether a Service Is Still Running

Alerta also offers a feature from the world of passive monitoring: heartbeats. A script or cron job must call this API endpoint regularly. If this call is not made, Alerta can throw an alert. This feature is useful for backup

What?!
I can get my issues SOONER?

Available anywhere, anytime!

Sign up for a digital subscription to improve your admin skills wth practical articles on network security, cloud computing, DevOps, HPC, storage and more!

Subscribe to the PDF edition:
shop.linuxnewmedia.com/shop
ADMIN
Network & Security
5 Network Admin Distros
Local AI Server
Snapshots in Azure
Checkmk

jobs, for example, so you can immediately see whether they are working properly. An existing backup script can be extended to include `heartbeat` in a `curl` command:

```
curl --request POST ➘
     --header 'Authorization: ➘
              Key H0i_bj<...>' ➘
     --header 'Content-type: ➘
              application/json' ➘
     --data '{ "origin": "gitea", ➘
              "timeout": 360 }' ➘
 https://<alerta.example.net>/api/heartbeat
```

Alerta displays the update messages in the Heartbeats section of the web GUI (**Figure 3**). Servers or services that do not phone home for a longer period of time are marked *expired*. Unfortunately, an alert is not generated automatically, but only after running the

```
alerta heartbeats --alerts
```

command. The documentation recommends setting up this command as a cron job that runs every five minutes.

Alerta Reports

Alerta can receive messages and can forward them to other Alerta servers. This feature is designed for environments with hierarchical structures. For example, the local Alerta server at a company site receives all the alerts and forwards them to company headquarters. The receiving server can even use a different alert manager, as long as the software can interpret the JSON messages correctly.

Another use case is a high-availability Alerta cluster: Alerta forwards incoming alarms from cluster node A to node B and vice versa, keeping the two servers synchronized, and it automatically rules out infinite loops.

The next reporting function relates to top 10 reports: Which services were most frequently affected? In the Reports section of the web GUI, Alerta lists the problem cases sorted by message count, which means you can deal with particularly disruptive candidates in a targeted way. For example, if the same server is complaining about a high CPU load every day at around four in the morning, the daily cron job could be a possible cause.

Finally, do not forget that the alerting software is open for changes and offers plenty of scope for integrations and minor tweaks. The developers provide a library for the Python programming language, making Alerta particularly easy to integrate into Python scripts. Also very appealing is the ability for third-party providers, such as AWS CloudWatch and Grafana, to use webhooks to connect to Alerta and generate alerts there.

Conclusions

Alerta is not a typical monitoring system that queries the network status; instead, it is a collection point for alerts from other monitoring systems. Alerta gains an overview of which monitoring environments are operating normally and where things are currently falling apart. Alerta comes with a command-line tool that lets you interact with your monitoring environment and can also be controlled entirely by the API, which means that integration with any programming language is possible. Alerta is also at home in the cloud, making it ideal for service providers. ■

Info

[1] Alerta: [https://alerta.io]

[2] Demo environment: [https://try.alerta.io/alerts]

[3] docker-alerta: [https://github.com/alerta/docker-alerta]

[4] Python Alerta client: [https://github.com/alerta/python-alerta-client]

[5] Nagios-to-Alerta gateway: [https://github.com/alerta/nagios-alerta]

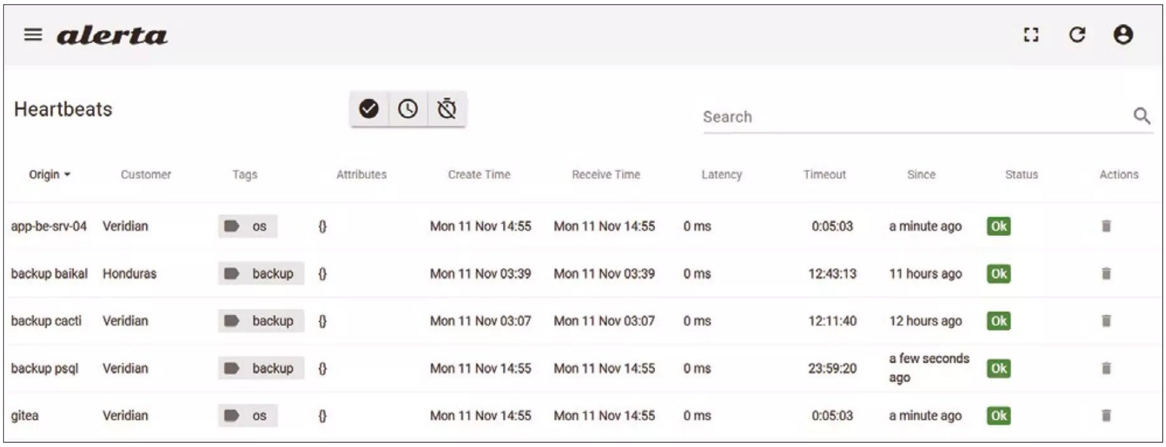
[6] Alerta documentation: [https://docs.alerta.io]

[7] Docker Compose file with Alerta, PostgreSQL, and Caddy: [https://www.it-administrator.de/sites/default/files/2025-02/docker-compose.txt]

[8] API kit: [explorer.alerta.io]

The Author

Markus Stubbig is a networking engineer who has worked in the IT industry for 20 years. His strong focus is on design and implementation of campus networks around the world.

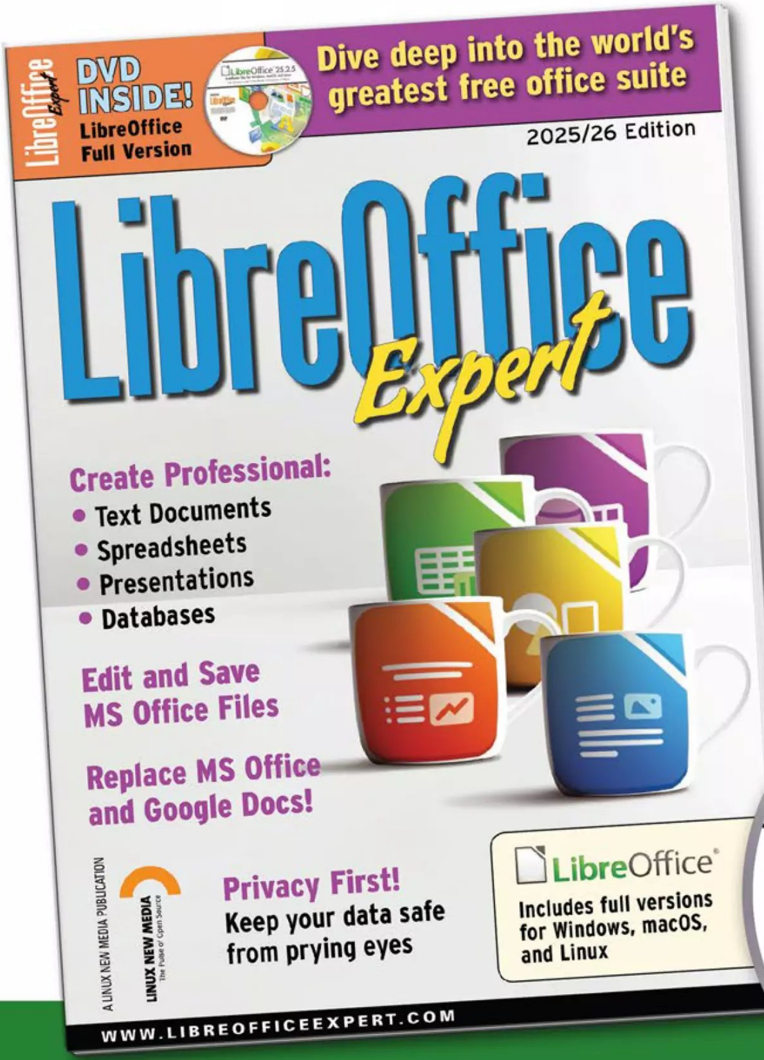


Origin ▾	Customer	Tags	Attributes	Create Time	Receive Time	Latency	Timeout	Since	Status	Actions
app-be-srv-04	Veridian	os	{}	Mon 11 Nov 14:55	Mon 11 Nov 14:55	0 ms	0:05:03	a minute ago	Ok	🗑
backup baikal	Honduras	backup	{}	Mon 11 Nov 03:39	Mon 11 Nov 03:39	0 ms	12:43:13	11 hours ago	Ok	🗑
backup cacti	Veridian	backup	{}	Mon 11 Nov 03:07	Mon 11 Nov 03:07	0 ms	12:11:40	12 hours ago	Ok	🗑
backup psql	Veridian	backup	{}	Mon 11 Nov 14:55	Mon 11 Nov 14:55	0 ms	23:59:20	a few seconds ago	Ok	🗑
gitea	Veridian	os	{}	Mon 11 Nov 14:55	Mon 11 Nov 14:55	0 ms	0:05:03	a minute ago	Ok	🗑

Figure 3: Heartbeats wait for regular signs of life. If no confirmation is received, an alert can be generated.

Shop the Shop
shop.linuxnewmedia.com/shop

Become a LibreOffice Expert

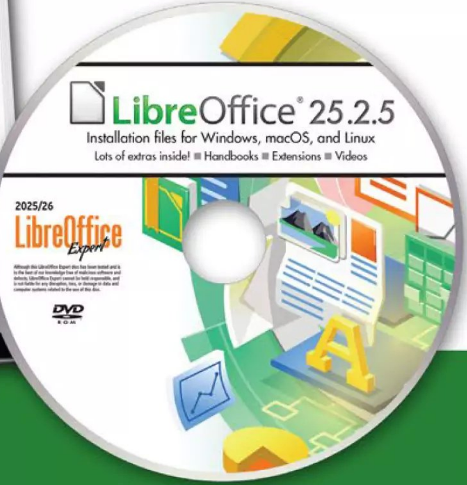

DVD INSIDE!
LibreOffice Full Version
Dive deep into the world's greatest free office suite
2025/26 Edition
LibreOffice Expert
Create Professional:
• Text Documents
• Spreadsheets
• Presentations
• Databases
Edit and Save MS Office Files
Replace MS Office and Google Docs!
Privacy First!
Keep your data safe from prying eyes
LibreOffice
Includes full versions for Windows, macOS, and Linux
A LINUX NEW MEDIA PUBLICATION
LINUX NEW MEDIA
The Pulse of Open Source
WWW.LIBREOFFICEEXPERT.COM

Explore the FREE office suite used by busy professionals around the world!

Create Professional:

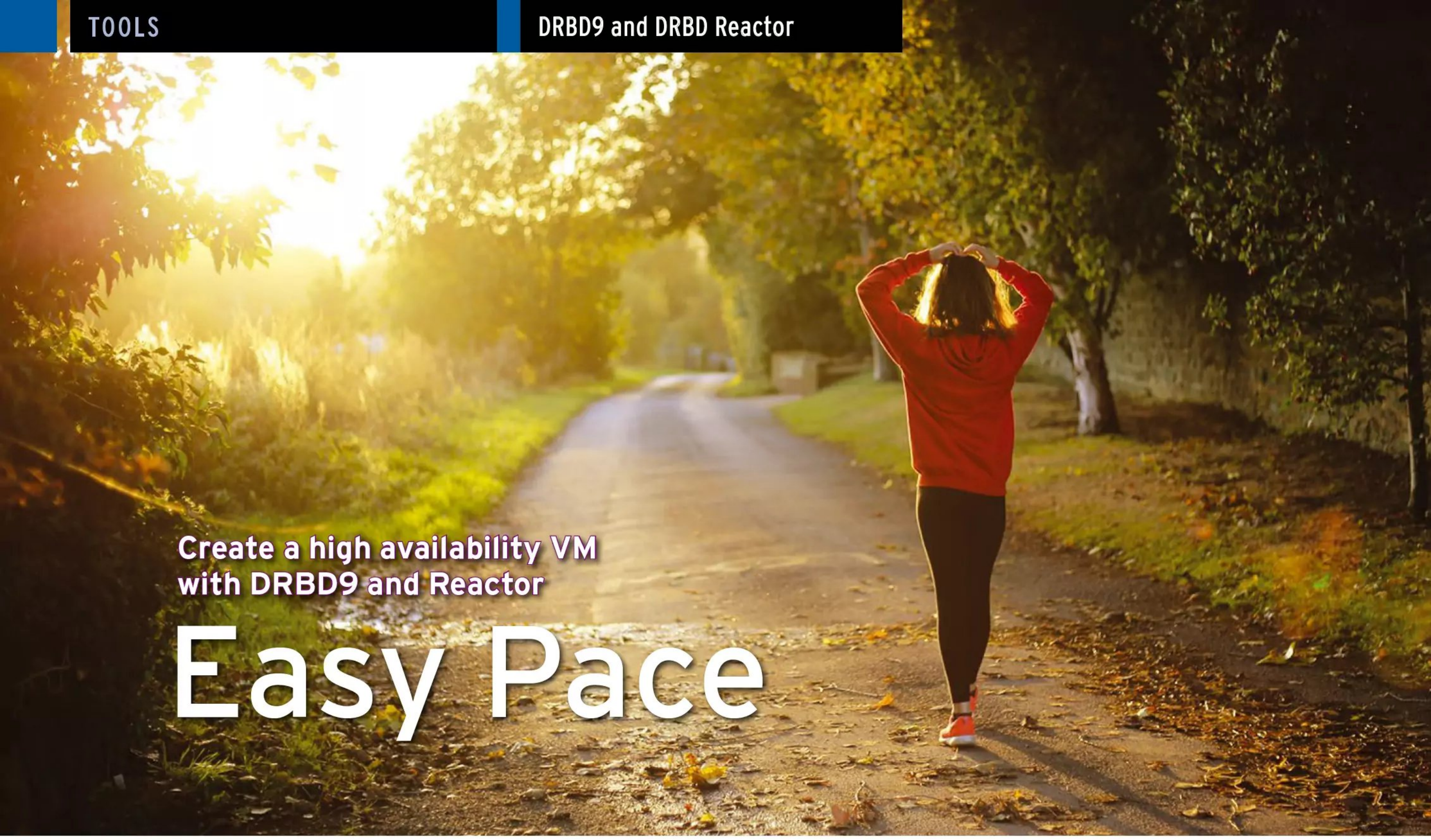
- Text Documents
- Spreadsheets
- Presentations
- Databases

Whether you work on a Windows PC, a Mac, or a Linux system, you have all you need to get started with LibreOffice today. This single-volume special edition will serve as your guide!


LibreOffice 25.2.5
Installation files for Windows, macOS, and Linux
Lots of extras inside! Handbooks Extensions Videos
2025/26
LibreOffice Expert
DVD

Order online:
shop.linuxnewmedia.com/shop

For Windows, macOS, and Linux users!



Create a high availability VM with DRBD9 and Reactor

Easy Pace

DRBD9 and DRBD Reactor create a Linux high-availability stack for virtual instances with replicated storage comparable to the classic Corosync and Pacemaker solution. By Martin Loschwitz

In today's IT landscape, everything has to scale: Bigger, better, more is the saying. However, this does not mean that classic high availability (HA) in the form of two-node clusters is a thing of the past. Usually, a legacy HA cluster steps in to ensure that central services required for platform operation survive failures and other issues. In some cases, the required services run directly on dedicated hardware with Linux, but it is far more common to assign the service to virtual instances and let a cluster manager manage them.

The combination of DRBD, Pacemaker, and KVM with Libvirt is established and reliable but not particularly popular, mainly because of the Linux HA stack (see the "Linux HA Stack" box). DRBD is a convenient alternative for most administrators. On many sites, you will find DRBD9 instead of DRBD8. The newer version is just as easy to use and configure in terms of basic configuration.

The crux of the matter is that the two-node scenario is still one of the most common in which Pacemaker is used. Many administrators only need a service that can initiate a failover in the typical early Heartbeat manner as soon as a system fails. Therefore, many setups do not benefit from most of the complexity in Pacemaker, although you still have to deal with it. Companies such as Linbit [5], which specializes in consulting for Linux HA, report that they repeatedly encounter setups with faulty Pacemaker configurations. Debugging these problems is time consuming and expensive, which is what prompted Linbit to build an alternative to Pacemaker that specializes in handling simple use cases: DRBD Reactor [6].

In this article, I describe how DRBD and DRBD Reactor can be used to implement a high-availability cluster, in which you can operate virtual machines with KVM and Libvirt. The Linux HA stack is largely ignored. DRBD Reactor only uses software from the stack in one place: Open Cluster Framework (OCF)-standard resource agents are used to manage the virtual machines from Reactor [7].

Keeping Things Simple

Linbit can rightly be accused of having an ulterior motive for developing Reactor: making its own core product, DRBD, more attractive to customers. RAID 1 over the network is already a good alternative to expensive network-attached storage (NAS) or storage area network (SAN), especially in smaller installations.

Another advantage is that Linbit releases DRBD under the terms of the free GPL license, which makes it far easier for companies to replace DRBD with an alternative solution than is the case with proprietary NAS or SAN appliances. However, Linbit has clearly identified Pacemaker as a major obstacle to the distribution of DRBD. It claims that its complexity prevents customers from building Linux HA solutions in the first place. Reactor is the Vienna-based company's attempt to replace Pacemaker with a less complex solution wherever possible.

Linbit expressly points out that Reactor is a standalone component that also works without DRBD. If a customer wants to use DRBD Reactor without DRBD, they can simply use an iSCSI volume or an NVMe-over-fabrics (NVMe-oF) target to connect shared storage to their target systems. However, the default use case intended by Linbit is likely to be that of combining DRBD9 and Reactor. The two components are designed for this purpose, which is why I look at this

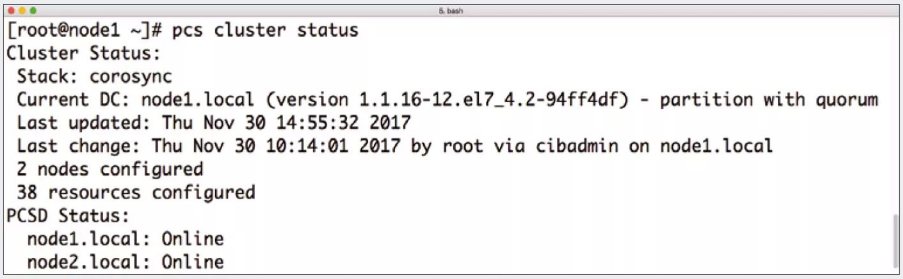
Photo by Emma Simpson on Unsplash

Linux HA Stack

Fifteen years ago, the world looked very different when it came to Linux and clusters. Heartbeat, developed primarily by Alan Robertson at IBM, was little more than a mess of shell scripts that could restart services on another server in the event of a server failure; then, the stack caught the attention of the major distributors. First, SUSE commissioned Andrew Beekhof to develop Heartbeat [1] into a full-fledged cluster manager, which became known as Heartbeat 2. Beekhof then joined Red Hat, where he was tasked with integrating various features into Heartbeat that would enable the operation of clusters with more than two nodes. Soon, the project changed its name to Pacemaker.

Pacemaker has become increasingly complex since then. Not only does it still rely on a configuration file under the hood that in turn relies on XML and is correspondingly complicated to use, the cluster stack has also been split into several components. First, the cluster communication layer was separated from the cluster manager and renamed Corosync [2], although many an annoyed admin refers to this component as Horrorsync.

Corosync was later split up again. Whereas Corosync now primarily handles the exchange of data between the different systems in a cluster, Kronosnet [3] ensures that the information is routed correctly across the network and arrives at the right destination. Pacemaker now has a Pacemaker configuration system (PCS) management daemon named PCSD [4] (Figure 1), all of which has made Pacemaker even more complex than it already was.



```
[root@node1 ~]# pcs cluster status
Cluster Status:
 Stack: corosync
 Current DC: node1.local (version 1.1.16-12.el7_4.2-94ff4df) - partition with quorum
 Last updated: Thu Nov 30 14:55:32 2017
 Last change: Thu Nov 30 10:14:01 2017 by root via cibadmin on node1.local
 2 nodes configured
 38 resources configured
PCSD Status:
  node1.local: Online
  node2.local: Online
```

Figure 1: The Linux HA stack has become increasingly complex in recent years – for example, with the introduction of the PCSD management daemon.

example as the default scenario in this article.

Easy to Get Started

The combination of DRBD and Reactor has become the standard approach for several reasons. To date, DRBD offers one of the simplest approaches to implementing shared storage in data centers. External add-on devices such as NAS or SAN mean sourcing and setup overhead, and they are usually far more expensive. Although replicated storage for a few terabytes can quickly cost tens of thousands of dollars, the same can often be achieved on a smaller scale with DRBD with a RAID controller and appropriately sized non-volatile memory devices (NVMs) in the machines that already form the HA cluster.

This arrangement makes the entry barrier for DRBD lower than for Ceph and is an open source-based solution, so you don't have to worry about the dreaded vendor lock-in. However, because of its implicit latency, Ceph often has problems in HA setups that need to provide fast storage for files or virtual instances. DRBD is comparatively simple under the hood, and you won't find any complex algorithms, unlike with Ceph, where CRUSH [8] sets the tone.

All told, the barriers to entry are lower for DRBD than for virtually any other storage solution. Moreover, packages of the current DRBD9 version are now available for most distributions, including one by the manufacturer Linbit itself, which makes it easy to build a DRBD cluster that can also run virtual instances from two off-the-shelf standard servers with the appropriate storage media.

DRBD is also well integrated with common IT solutions. A native interface to Kubernetes exists, as does a connection to OpenStack. Linbit relies on DRBDManage, the management framework built specifically for DRBD9. A community edition is also available, without any major restrictions, and is free of charge as a package for all common distributions.

Installing the Packages

Assume you have two freshly installed systems with Ubuntu 24.04, installed either manually or with bare metal lifecycle management and automation. The only important thing is that both systems have the latest software, which is the precondition for integrating the community directory [9] for DRBD9 by Linbit into the system. The steps required follow the standard procedure for integrating external package sources into Ubuntu: Store the manufacturer's community directory on the system and enable it; then, download the packages for DRBD9 to the system:

```
$ sudo add-apt-repository ➘
  ppa:linbit/linbit-drbd9-stack
$ sudo apt update
$ sudo apt install drbd-dkms drbd-utils ➘
  python-drbdmanage drbd-reactor
```

Because the manufacturer provides DRBD9 in the form of a DKMS package, DKMS automatically creates the required kernel module. Depending on the configuration, you might need to run

```
modprobe drbd
```

at the command line. The `dmesg` command should then display a message telling you that the DRBD module is now loaded. Besides DRBD, the `apt` command has also downloaded its management tools and DRBDManage to the system; DRBD Reactor is also installed.

Setting up a DRBD Resource

One kernel module does not a HA stack make. The administrator must use the DRBD module to set up a replicated storage drive. The example assumes that a storage drive `/dev/sdb/` exists on both systems of the future HA cluster (named *alice* and *bob*), which forms the basis of the future HA cluster. Under the hood, this could be the RAID drive described above, a RAID 5, or a RAID 6 consisting of several NVMs.

One challenge with this type of setup is that the drives are likely to be several terabytes in size, but a single VM usually does not need that much storage space. This problem can be circumvented by Linux logical volume management (LVM) [10]. It is then important that the respective logical volume (LV) is used as the back-end drive for DRBD and that replication also takes place on the basis of these LVs. The setup differs slightly from the standard procedure used in the past. Previously, it was quite common to make LVM itself a cluster resource and manage it with Pacemaker. However, this arrangement is more complicated than necessary.

Listing 1: DRBD Resource for VM

```
# vi /etc/drbd.d/proxy.res
resource proxy {
  device drbd0;
  disk /dev/drbd/proxy;
  meta-disk internal;
  on alice {
    address 10.42.0.100:7789;
    node-id 0;
  }
  on bob {
    address 10.42.0.101:7789;
    node-id 1;
  }
  connection-mesh {
    hosts alice bob;
  }
  options {
    auto-promote no;
    quorum majority;
    on-suspended-primary-outdated force-secondary;
    # for the default drbd-reactor
    # on-quorum-loss policy (i.e. Shutdown)
    on-no-quorum io-error;
    # always set this to the value of on-no-quorum!
    on-no-data-accessible io-error;
  }
  net {
    protocol C;
    after-sb-0pri discard-zero-changes;
    after-sb-1pri discard-secondary;
    after-sb-2pri disconnect;
  }
}
```

The next step is to set up the data carriers for DRBD, which can be done in Ubuntu 24.04 by typing:

```
pvcreate /dev/sdb
vgcreate drbd /dev/sdb
```

The first command creates the metadata required for LVM on the drive, and the second command creates a volume group named `drbd` on both systems and adds the previously created physical volume to it.

Now you can create the first logical volume. On both systems, the command

```
lvcreate -L100G -n proxy drbd
```

handles everything you need. This example sets up a 100GB volume named `proxy` for later use as storage for a VM running a Squid proxy server.

Next, you need to set up the DRBD resource. The configuration file from Listing 1 is fine for this example, but it could also be easily modified for local use.

If the Ubuntu systems are rolled out with an active firewall, such as UFW, the specified port – 7789 here – must be allowed. To create the metadata for the resource on both systems, use:

```
drbdadm create-md proxy
drbdadm up proxy
```

Finally, you need to define one of the drives as the canonical source for data accuracy. Because the drives are empty anyway, it does not matter whether you run the

```
drbdadm primary proxy --force
```

command on *alice* or *bob*.

Setting Up a Virtual Instance

From an administrative point of view, the next step is just part of the admin's daily grind and has little to do with DRBD. The virtual instance needs to be installed. To do this, simply install KVM and Qemu on both systems, enable the Intel

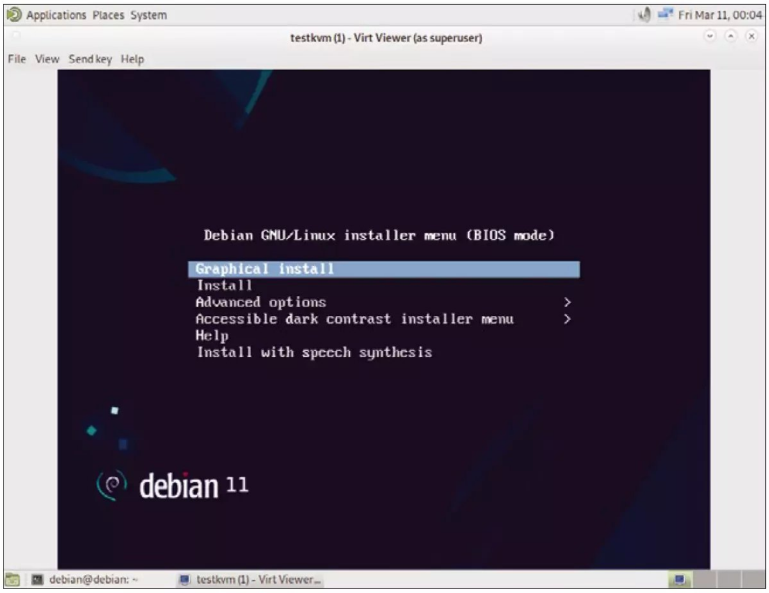

Applications Places System
Fri Mar 11, 00:04
testkvm (1) - Virt Viewer (as superuser)
File View Send key Help
Debian GNU/Linux installer menu (BIOS mode)
Graphical install
Install
Advanced options >
Accessible dark contrast installer menu >
Help
Install with speech synthesis
debian 11
debian@debian: ~
testkvm (1) - Virt Viewer...

Figure 2: The `virt-install` command lets you install a virtual instance at the command line. Linbit `virter` is one possible alternative. © Debian Handbook, CC BY-SA 3.0 [11]

virtualization technology (VT) or AMD secure virtual machine (SVM) flag for the respective CPU in the BIOS, and install Libvirt, which then handles the management of the instances. To roll out the virtual instance on the node where the DRBD resource was previously switched to `primary` mode, use `virt-install`:

```
$ virt-install ↩
  --name ubuntu2404 --ram 4096 ↩
  --disk path=/dev/drbd/proxy --vcpus 2 ↩
  --os-variant ubuntu24.04 ↩
  --network bridge=br0 --graphics none ↩
  --console pty,target_type=serial ↩
  --location /tmp/ubuntu-24.04-live-↩
             server-amd64.iso, ↩
    kernel=casper/vmlinuz,↩
    initrd=casper/initrd ↩
  --extra-args 'console=ttyS0,115200n8'
```

This command is intended as an example. It assumes that a network bridge named `br0` is set up on the system and that an ISO image of Ubuntu 24.04 Server is located in `/tmp`. Additionally, the example contains instructions for attaching a virtual serial console to the instance; its contents are immediately displayed on the local terminal (**Figure 2**).

It makes sense to select the *Continue in basic mode* option and then continue with the installation in the usual way. Once this completes, `/dev/drbd/proxy/` should contain a complete installation of Ubuntu 24.04, which is mirrored on the cluster partner.

It is important to copy the configuration file from the instance (`/etc/libvirt/qemu/ubuntu2404.xml`) to the same location on the other cluster system and then restart Libvirt by typing

```
systemctl restart libvirt-bin
```

Linbit itself offers the `virter` tool, which can create and clone Libvirt VMs over the command-line interface [12].

Quick Start

Strictly speaking, DRBD Reactor itself is not a classic cluster manager. Many of the functions that Pacemaker takes from Corosync and Kronosnet are not found in Reactor, which does not mean that the functions are missing; instead, DRBD Reactor relies on functionality that is already available in systemd and DRBD. Most of the power in the DRBD Reactor promoter plugin lies in creating systemd units: It sets up the systemd configuration so that the tool responds in a defined manner when certain events occur. To do so, DRBD Reactor primarily listens on the system bus and records the events it detects. If an event for which a DRBD Reactor rule has been created passes through the system bus, Reactor sets the respective DRBD resource to the desired state, if configured accordingly, and then lets systemd do most of the rest of the work. Therefore, the configuration required for DRBD Reactor in the example is quite straightforward.

Save the lines

```
[[promoter]]
[promoter.resources.proxy]
start = ["ocf:heartbeat:VirtualDomain ↩
          p_kvm-proxy config=/etc/↩
          libvirt/qemu/proxy.xml"]
```

in `/etc/drbd-reactor.d/proxy.toml`; then,

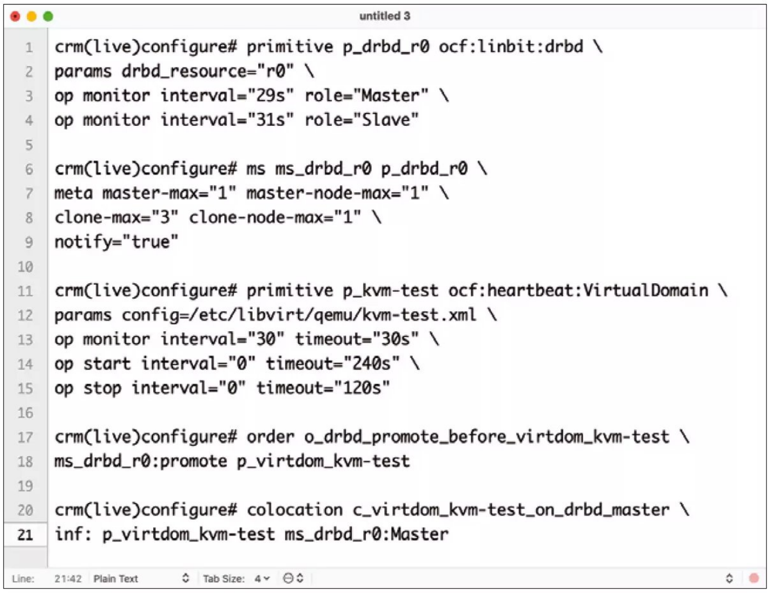
```
systemctl restart drbd-reactor
```

ensures the service uses the new configuration. It is worth noting how much less complex this setup is compared with Pacemaker (**Figure 3**).

Under the Hood

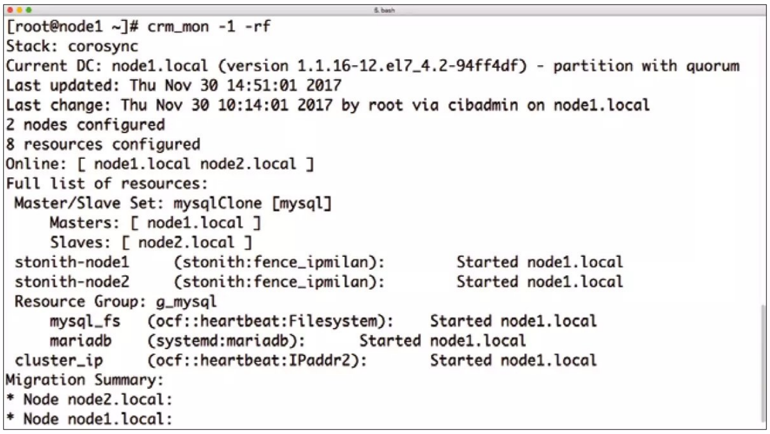
Anyone who has worked with DRBD before will probably wonder how the automatic processes looked at here work with regard to DRBD itself. If you have already set up DRBD in conjunction with Pacemaker, you will be aware that it is not entirely trivial. Depending on the setup, Pacemaker not only needs to take care of the actual DRBD resource, but also of any neighboring resources. If DRBD is to act as the underpinnings for a Samba server, Pacemaker needs to make sure, for example, that the cluster IP and the mount point of the DRBD resource's filesystem are changed whenever the primary role of the DRBD resource changes.

The OCF resource agent `ocf:heartbeat:Filesystem` helps, but don't forget the DRBD resource itself. In the basic configuration, it can only be used if it is in primary mode. This part of the setup considerably adds



```
crm(live)configure# primitive p_drbd_r0 ocf:linbit:drbd \
params drbd_resource="r0" \
op monitor interval="29s" role="Master" \
op monitor interval="31s" role="Slave"

crm(live)configure# ms ms_drbd_r0 p_drbd_r0 \
meta master-max="1" master-node-max="1" \
clone-max="3" clone-node-max="1" \
notify="true"

crm(live)configure# primitive p_kvm-test ocf:heartbeat:VirtualDomain \
params config=/etc/libvirt/qemu/kvm-test.xml \
op monitor interval="30" timeout="30s" \
op start interval="0" timeout="240s" \
op stop interval="0" timeout="120s"

crm(live)configure# order o_drbd_promote_before_virtdom_kvm-test \
ms_drbd_r0:promote p_virtdom_kvm-test

crm(live)configure# colocation c_virtdom_kvm-test_on_drbd_master \
inf: p_virtdom_kvm-test ms_drbd_r0:Master
```

Figure 3: **In Pacemaker, the configuration shown here would be needed to realize the same effect that the DRBD Reactor TOML configuration achieves.**



```
[root@node1 ~]# crm_mon -1 -rf
Stack: corosync
Current DC: node1.local (version 1.1.16-12.el7_4.2-94ff4df) - partition with quorum
Last updated: Thu Nov 30 14:51:01 2017
Last change: Thu Nov 30 10:14:01 2017 by root via cibadmin on node1.local
2 nodes configured
8 resources configured
Online: [ node1.local node2.local ]
Full list of resources:
 Master/Slave Set: mysqlClone [mysql]
     Masters: [ node1.local ]
     Slaves: [ node2.local ]
 stonith-node1      (stonith:fence_ipmilan):        Started node1.local
 stonith-node2      (stonith:fence_ipmilan):        Started node1.local
 Resource Group: g_mysql
     mysql_fs   (ocf::heartbeat:Filesystem):    Started node1.local
     mariadb    (systemd:mariadb):      Started node1.local
 cluster_ip     (ocf::heartbeat:IPaddr2):       Started node1.local
Migration Summary:
* Node node2.local:
* Node node1.local:
```

Figure 4: Setups that manage resources such as DRBD in the cluster are now specified as in promoted (master) or unpromoted (slave) states.

to the complexity of the configuration in Pacemaker, which has its own construct for distributed services running in the cluster that have different states for different instances across the cluster.

Such is the case with DRBD: If the resource is in use, it is running in primary mode. If the node is only receiving updates, it is in secondary mode and cannot be used on the system itself. The DRBD kernel module will respond with an error message to access attempts.

In DRBD9, the situation is even more complex with regard to Pacemaker, which natively supports more than two nodes. Pacemaker recognizes promoted and unpromoted states (previously designated master-slave; Figure 4) for multinode services. This feature can be quite annoying for administrators, because they are complicated to configure. This complexity is true for two nodes, and even more so for multinode installations. Linbit states that a significant portion of the problems encountered by DRBD customers in everyday use are related to these configuration instructions.

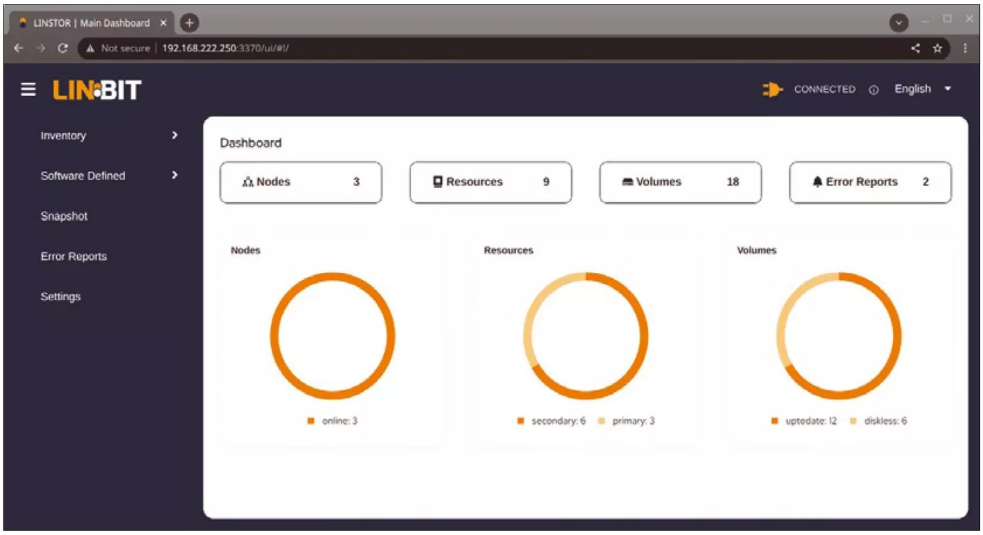


Figure 5: The Linstor GUI manages DRBD clusters graphically on request and, by deploying an arbiter node, takes care of the quorum issue, which can be a pain for admins. © Linbit

Linbit addressed this issue some time ago and added a feature to DRBD9 that supports automatic activation of the primary mode for a resource where access is required. The construct has some security features: DRBD9 does not attempt to switch a resource to active mode automatically if it is already active elsewhere in the cluster. This feature is important to avoid split-brain situations. DRBD Reactor benefits greatly from automatic switching to promoted mode, because this feature is a central component of its functionality.

Further Considerations

If you are looking to manage DRBD9 with Linstor [13], Linbit's software-defined storage (SDS) application under a free license, nothing can stop you from doing so. The services are mutually integrated and collaborate well. Linstor (Figure 5) also solves a problem that you would otherwise need to tackle elsewhere: It implements a quorum. This issue is sensitive with two-node clusters, because it is essential, if the two cluster nodes lose their connection to each other, to prevent both from considering themselves the remaining active node and activating all services.

Pacemaker uses the intelligent platform management interface (IPMI) and Shoot the Other Node in the Head (STONITH) service to carry out a number of tasks and help avoid such a situation. In two-node clusters, however, the quorum is always a compromise, which is why Linstor lets you set up an arbiter node, also known as a quorum tiebreaker. It monitors both systems and is responsible for ensuring a quorum if the two cluster servers can no longer see each other. The hardware requirements for an arbiter are low; a Raspberry Pi is usually fine. If you have a free port in your rack, this option is far better than the typical Pacemaker workarounds.

Conclusions

Neither DRBD Reactor nor this article aim to denigrate the quality or practicality of the Linux HA stack or Pacemaker. Quite the contrary: Beekhof and the team of developers working on the various Linux HA components have spent the last decade turning a DIY solution into an enterprise-ready cluster manager. Today, clusters on Linux are no longer considered a makeshift solution. Whereas Linux administrators used to cast envious glances at AIX or HP-UX, which had comprehensive cluster managers on board at an early stage, the situation is now often reversed. As is so often the case, the open source community has done a great job in the field of clustering.

That said, Pacemaker has become so versatile and complex that admins are over their heads in many places. For simple setups especially, such as two-node clusters for running a few instances, admins often are not willing to familiarize themselves with a complex beast like Pacemaker. Instead, people just want the software to work and do the right thing on its own. At Linbit, they are trying to achieve exactly that outcome with DRBD Reactor.

DRBD Reactor is far less complex than Pacemaker. It avoids reliance on a multitude of components that need to interact flawlessly. You don't have to deal with different incompatible command-line interface tools that depend on which distribution you use (`pcs` on Red Hat, `crm` on SUSE). Instead, all you need is to create a DRBD resource and connect it to DRBD Reactor with a simple configuration file, which you can do indirectly with a virtual instance. This approach is noticeably easier than the Pacemaker method and ultimately leads to comparable results.

All of the components I looked at here are available free of charge online and work perfectly on Ubuntu 24.04. They are also published under a free license. A solution such as that outlined in this article also avoids vendor lock-in. It doesn't always have to be Pacemaker, and sometimes less can be more, as DRBD Reactor just goes to prove.

Info

[1] **Heartbeat:** [https://wiki.ubuntu.com/UbuntuHighAvailabilityTeam/Heartbeat]
[2] **Corosync:** [https://corosync.github.io/corosync]
[3] **Kronosnet:** [https://kronosnet.org]
[4] **PCSD:** [https://manpages.ubuntu.com/manpages/focal/man8/pcsd.8.html]
[5] **Linbit:** [https://linbit.com]
[6] **DRBD Reactor:** [https://github.com/LINBIT/drbd-reactor]
[7] **OCF resource agents:** [https://github.com/ClusterLabs/resource-agents]
[8] **CRUSH:** [https://docs.ceph.com/en/reef/rados/operations/crush-map]
[9] **DRBD9-PPA:** [https://launchpad.net/~linbit/+archive/ubuntu/linbit-drbd9-stack]
[10] **LVM:** [https://sourceware.org/lvm2]
[11] **Attribution-ShareAlike 3.0 Unported:** [https://creativecommons.org/licenses/by-sa/3.0/deed.en]
[12] **virter:** [https://github.com/LINBIT/virter]
[13] **Linstor:** [https://github.com/LINBIT/linstor-server]

The Author

Martin Loschwitz is the founder and managing director of True West IT Services GmbH, which offers scalable IT infrastructure based on OpenStack and Kubernetes.

Server validation with Goss

Checkup

Goss checks your servers, delivers a machine health status report, and allows you to evaluate the test results from a full-blown monitoring system. We show you how to get started with this free tool and how to run a variety checks on Linux and Windows computers. By Markus Stubbig

Goss describes itself as a tool for "quick and easy server validation" [1], which perfectly sums up its functionality. Configuring Goss is a breeze, and the checks are usually completed within a second. Plus, the financial outlay is zero thanks to the Apache open source license. On the downside, though, all the software really offers is unit testing (aka module testing or component testing). If a check fails, you will need to carry out an in-depth investigation to determine the cause. Goss has been testing Linux systems for almost 10 years, with support for Windows being added in 2020, although the Windows side still has some limitations (more on that later). In Docker, Goss runs its checks inside the container and outputs a report. Goss is not the first software of its kind and sees itself as an alternative to Serverspec, a tool that is more versatile but also more complex to configure than Goss.

Getting Goss Up and Running

The `goss` command is all you need for a server check. Use either of the following commands:

```
goss validate
goss v
```

to instruct goss to examine the goss.yaml configuration file and begin the validation process. For impatient admins, in the next section I look at the `autoadd` function, which populates the configuration file with routine tasks. The Goss binary file is available from GitHub. Just download the file to your Windows or Linux computer from any web browser or at the Linux command line:

```
curl --location --output /usr/bin/goss ➘
  https://github.com/goss-org/goss/➘
  releases/download/v0.4.9/goss-linux-amd64
chmod +x /usr/bin/goss
```

The process is similar in Windows, where the `curl` command has been available since Windows 10:

```
curl --location --output %LOCALAPPDATA%\➘
  Microsoft\WindowsApps\goss.exe ➘
  https://github.com/goss-org/goss/➘
  releases/download/v0.4.9/➘
  goss-windows-amd64.exe
```

If the installation worked, a simple `goss -v` outputs the version number of the new software. Goss has no

Lead Image © Ewa Walicka, fotolia.com

graphical user interface, not even on Windows.

Auto-Configuration

Goss can compile some checks independently, including Linux services such as SSH and the Apache HTTPD server, and enter them in its configuration file. Goss does not generate a static configuration but composes the YAML lines to match the Linux host. For example, a customized checklist for the MongoDB database is created by running the command

```
goss autoadd mongod
```

which adds TCP port 27017, the `mongod` service, the service user, the group, and the process.
If you do not need all of these components, you can use individual `add` commands for a more targeted approach:

```
goss add service mongod
goss add user mongod
goss add port 27017
```

A quick look at the `goss.yaml` file tells you what Goss has created in a more-or-less readable format. The first command fills the configuration file with a `port` command and the associated options, which tells Goss to check whether a program is listening on TCP port 27017:

```
port:
  tcp:27017:
    listening: true
    ip:
    - 127.0.0.1
```

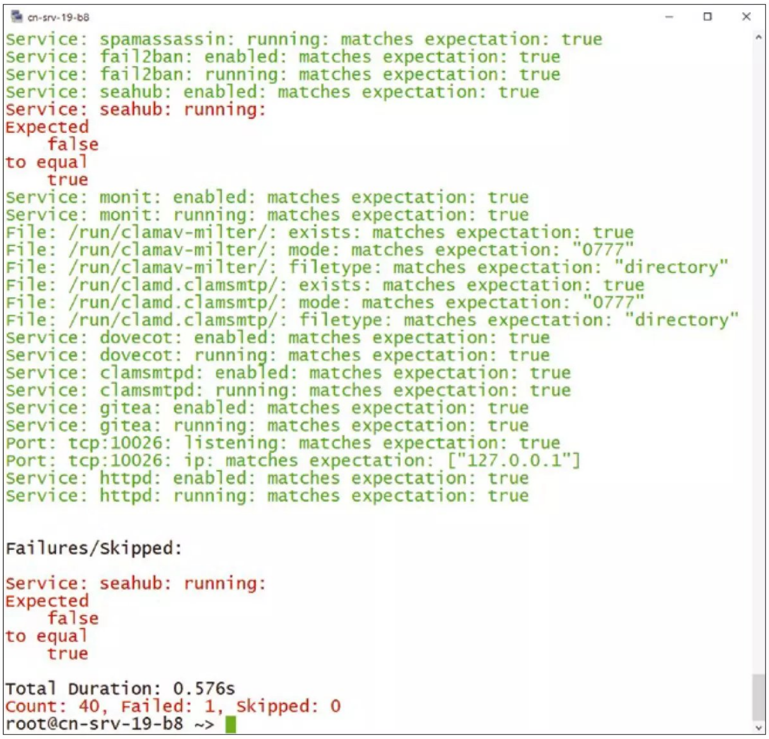
The `goss.yaml` file can be edited manually. The documentation [2] details further options for adding your own commands, checking name resolution, or querying kernel parameters. Many methods are not yet implemented in Windows, but new ones are added with each version.

Manual and Scripted Checks

A few initial checks are now complete and Goss can start testing. You can launch the checks by typing:

```
goss validate
```

You should see a few lines flash across the screen; after a few milliseconds, the results will be available. A successful overall result is displayed in green. If any tests fail, the software highlights the components that deviate from the target state in red (Figure 1).



```
Service: spamassassin: running: matches expectation: true
Service: fail2ban: enabled: matches expectation: true
Service: fail2ban: running: matches expectation: true
Service: seahub: enabled: matches expectation: true
Service: seahub: running:
Expected
    false
to equal
    true
Service: monit: enabled: matches expectation: true
Service: monit: running: matches expectation: true
File: /run/clamav-milter/: exists: matches expectation: true
File: /run/clamav-milter/: mode: matches expectation: "0777"
File: /run/clamav-milter/: filetype: matches expectation: "directory"
File: /run/clamd.clamsmtp/: exists: matches expectation: true
File: /run/clamd.clamsmtp/: mode: matches expectation: "0777"
File: /run/clamd.clamsmtp/: filetype: matches expectation: "directory"
Service: dovecot: enabled: matches expectation: true
Service: dovecot: running: matches expectation: true
Service: clamsmtpd: enabled: matches expectation: true
Service: clamsmtpd: running: matches expectation: true
Service: gitea: enabled: matches expectation: true
Service: gitea: running: matches expectation: true
Port: tcp:10026: listening: matches expectation: true
Port: tcp:10026: ip: matches expectation: ["127.0.0.1"]
Service: httpd: enabled: matches expectation: true
Service: httpd: running: matches expectation: true


Failures/Skipped:

Service: seahub: running:
Expected
    false
to equal
    true

Total Duration: 0.576s
Count: 40, Failed: 1, Skipped: 0
root@cn-srv-19-b8 ~>
```

Figure 1: If you want more detail, Goss can tell you what it expects for each check and highlight any items that fell short.

Listing 1: Initial Checks with Goss

```
command:
  mariadb:
    exit-status: 0
    exec: mysql cacti -s <<<'SELECT hostname FROM host
          WHERE id=1;'
    stdout:
    - localhost
  searchd index:
    exit-status: 0
    exec: /usr/bin/indextool --check note1
    stdout:
    - "check passed"
  private key length:
    exit-status: 0
    exec: "openssl rsa -in /www/tls/priv.key -text"
    stdout:
    - "/Private-Key: \\((2048|3072|4096) bit/"
```

To receive feedback automatically from the `goss validate` command when logging in over SSH, you will need to enter the command in your personal `.profile` file to ensure that Goss only provides feedback during interactive logins and not during SCP-based data transfer or before a `git push` over SSH.
Not all Linux statuses can be queried with the Goss onboard tools. For example, the tool detects whether the MariaDB process is running and listening on a specific network port, but it is not a clear indication that the database is working reliably. More detailed checks can be added as external commands or scripts for execution by Goss. Once the external command has been processed, Goss keeps track of the exit code and the program output. If the exit code shows the expected value and the output shows the desired pattern, the call is flagged as successful.
Listing 1 shows Goss calling arbitrary commands or scripts and then

Listing 2: Results from Multiple Servers

```
gossboss healthzs --servers "http://roc-sv-72-r:8080/
   healthz" --servers "http://dis-sx-01-d:8080/healthz"
   --servers "http://teg-sv-09-b:8080/healthz"

✓ http://dis-sx-01-d:8080/healthz

✗ http://roc-sv-72-r:8080/healthz

✓ http://teg-sv-09-b:8080/healthz

Error: Goss test failed
```

checking the exit code and command output. A `SELECT` command retrieves a value from MariaDB and compares it with a fixed character string. The Sphinx search engine is up next. The software sees the *check passed* output and deduces that the indexer is OK. Finally, the tool inspects a private key and uses a regex to compare its length against acceptable values.

Note that the commands must run without a password. Keep in mind that external scripts can significantly slow down the execution time mentioned earlier. If the script first spends a long time interacting with the database, Goss will not be able to return its overall verdict after just a few milliseconds.

Remote Checks

Goss checks the local machine, of course, but it can also address other servers on the network. The `addr` method attempts to establish a TCP connection to a remote host in a style similar to Telnet. In the DNS test, the tool resolves a name on an arbitrary DNS server (much like `nslookup` and `dig`). If the service is accessible and provides the desired response, Goss logs the check as successful.

For a web server, the `http` test can check availability, but also headers and bodies for specific content. The process is similar to that of the `curl` and `wget` tools. Goss also uses this approach to validate access to an HTTP API. For example, the application passes an IPv4 address to the API website and expects the correct geographical assignment:

```
http:
  geoip-api:
    status: 200
    url: http://geoip.example.net/1.2.3.4
    body:
    - Australia
```

Interestingly, no test equivalent to `ping` exists that sends a simple ICMP echo packet.

Interaction with Other Tools

Although Goss is a standalone tool, it integrates nicely with other monitoring tools. The `--format` switch lets you format the results such that Nagios or Prometheus can understand them. In Figure 1, Goss has detected a faulty service and color-highlighted it in the `documentation` format. Output in JSON format is also useful. At first glance, it is confusing because Goss outputs all the information without line breaks, but `jq` helps you restore order and pick out individual values.

For remote access, Goss has a built-in minimalist web server. Typing

```
goss serve
```

tells the tool to make its findings available on TCP port 8080 on the network. A monitoring system can access *http://server:8080/healthz* and query the results. Besides the visual display, the Goss server returns an HTTP status code of *200* for successful checks. If at least one check discovers errors, the code response is *503*.

Parallel Checks

Another tool in the Goss ecosystem is `gossboss`, a command that calls up multiple Goss instances and displays their statuses in plain text on the command line. A precompiled version of the tool as a tarball is available from GitHub [3]. The installation follows the same steps as for Goss.

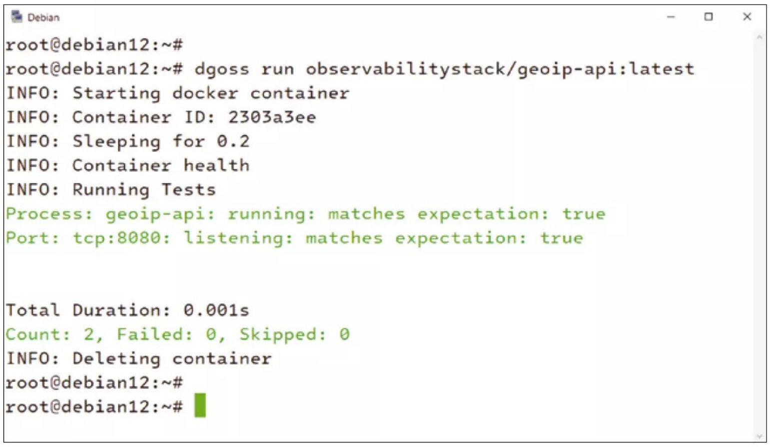
Gossboss uses the `serve` function for the above-mentioned functionality. Because Gossboss requires output in JSON format, Goss must be run with

```
goss serve --format json
```

on the servers involved. Gossboss then asks its clients to report in one after the other (Listing 2).

Container Images

Goss not only checks the host, but also its guests, which are located in containers. The command you need, `dgoss`, comes from the same provider and installs in the usual way [4]. The command also expects a check file in YAML format and a Docker image. The `dgoss` command then starts a container, works through its checklist, stops the container, and presents the results (Figure 2), which means you are



```
root@debian12:~#
root@debian12:~# dgoss run observabilitystack/geoip-api:latest
INFO: Starting docker container
INFO: Container ID: 2303a3ee
INFO: Sleeping for 0.2
INFO: Container health
INFO: Running Tests
Process: geoip-api: running: matches expectation: true
Port: tcp:8080: listening: matches expectation: true


Total Duration: 0.001s
Count: 2, Failed: 0, Skipped: 0
INFO: Deleting container
root@debian12:~#
root@debian12:~#
```

Figure 2: The dgoss variant can be used to check Docker containers.

checking the functionality of the Docker image and not of a running container.

The `dcgoss` command, which is designed for Docker Compose environments, works in the same way, whereas `kgoss` takes Goss into the world of Kubernetes pods.

Security

Goss not only checks whether certain files are present and services are running, but also does the opposite. For example, if Sendmail is an undesirable entity in your organization, Goss can make sure that no package of that name is installed and that no processes with *sendmail* in their titles are running. In-depth security audits are possible with Goss, but they are time-consuming. Tools such as Lynis are faster for this purpose.

Conclusions

Goss performs quick checks on servers. The individual checks can be created in YAML format and used to check files, processes, network ports, services, and kernel parameters. Goss is useful for quick checks before a planned configuration change and after an update. The tool can format the results for Nagios to support integration with a monitoring system. At the end of the day, though, Goss is not a full-fledged monitoring tool, because it lacks the ability to restart faulty services automatically, to cite just one example.

Info

[1] Goss: [https://github.com/goss-org/goss]
[2] Goss documentation: [https://goss.readthedocs.io/en/stable/]
[3] Gossboss: [https://github.com/mdb/gossboss]
[4] dgoss: [https://github.com/goss-org/goss/tree/master/extras/dgoss]

The Author

Markus Stubbig is a networking engineer who has worked in the IT industry for 20 years. His strong focus is on design and implementation of campus networks around the world.

IT Highlights at a Glance

Too busy to wade through press releases and chatty tech news sites? Let us deliver the most relevant news, technical articles, and tool tips – straight to your Inbox.

ADMIN Update • **Linux Update**

Keep your finger on the pulse of the IT industry.

ADMIN Update: bit.ly/ADMIN-Update
Linux Update: bit.ly/Linux-Update

Photo by Andrew Neel on Unsplash

MIND THE GAP

Windows updates and patch distribution without WSUS

Clear Signs

Microsoft decided to discontinue the Windows Server Update Services update helper and declare it deprecated at the end of Windows 2025 support. We look at whether companies need to take action now and analyze the tools available as alternatives for Windows updates and patch distribution.

By Tam Hanna

Microsoft Windows Server Update Services (WSUS) server role has established itself as an effective tool for managing Windows updates. However, if you take a closer look at the Windows ecosystem, the decision to discontinue WSUS was to be expected – the trend toward the cloud is clear. Moreover, Microsoft has not actively developed the features implemented in WSUS for some time now.

No Immediate End

Discontinuation announcements often trigger panic reactions in the Microsoft world, but if practical experience with Windows technologies teaches us one thing, it is that talk is often greater than the effect. The best example of this is the fuss surrounding the discontinuation of Visual Basic 6 (VB6); despite all the doomsday scenarios, VB6-based applications still run without issues on Windows 11. As a general rule, you can say that Microsoft is reluctant to discontinue services that are actively requested by its customers.

The most important fact concerning WSUS is that Microsoft has indicated in its WSUS roadmap that the product will be supported by Windows Server 2025, which means a long update cycle: Microsoft has set the end date for mainstream support – five years, as always – to October 9, 2029. Extended commercial support with security updates will be available for another five years; that is, until 2034. The prices for this extended service vary from case to case and depend not least on an organization's negotiating skills. You can find cost estimates online [1] that are based on an annual price of up to 75 percent of the original license charge.

Action will need to be taken by October 2029 at the latest, but because this date is still quite a long way off, you currently have no reason to take any action. Another possibility is that Microsoft will change its deprecation decision and integrate the service into a successor version.

Luring Users into the Cloud

Of course, Microsoft would prefer you move your entire IT infrastructure to the Azure cloud. The Azure Update Manager service takes care of automatically maintaining resources hosted in the cloud (Figure 1). Thanks to its integration with the Azure Arc hybrid cloud service, you can also use Azure Update Manager to manage local servers. The only prerequisites are Arc integration and willingness to pay the price. Besides Windows, Microsoft also supports Linux versions commonly used in the enterprise sector with Azure Update Manager [2].

One very practical feature of Azure Update Manager is that its integration into the local network does not require a specific setup. Anyone who uses an Azure-hosted Windows instance or integrates their Windows server into Azure Arc automatically receives the update functions free of charge.

If you have a virtual machine hosted in Azure, the service is even free of charge, whereas on-premises use costs up to $5 per month, depending on the machine's uptime. However, note that a single connection will trigger billing for the entire day.

Once integrated, updates can be installed on individual hosts directly from the Azure back end. However, manual update delivery is only one part of overall functionality. With the use of a wide range of group policy guidelines, Azure Update Manager lets you configure triggers to deliver updates automatically across your entire fleet.

Versatility

All told, Azure Update Manager is flexible and multifaceted. For example, the periodic assessment feature can be used to automate monitoring of the states of various virtual machines (VMs). In this case, the system carries out periodic checks to report missing updates or install them directly.

Integration with the Azure Event Grid is also worth mentioning in the context of automation. While an update is executing, various events can be processed or triggered. A classic example would be booting additional machines during an update phase, either to avoid

Photo by Rob Thompson on Unsplash

bottlenecks or to ensure that all machines are updated simultaneously. You should note that the execution time of the various payloads is subject to fairly strict limits, so complex computations or interaction with external servers should be handled asynchronously.

To sum up, Azure Update Manager really comes into its own when an organization's IT infrastructure is already in the Azure cloud. A load balancer could then, for example, automatically respond to the reduction in compute power caused by the installation process. VMs hosted in Azure do not typically cause any major problems in terms of transfer costs for updates. With a locally based infrastructure, though, it might be advisable to reduce the transfer volume by using a local cache. One example of this solution is Microsoft Connected Cache [3], which is currently in beta.

Windows Autopatch

Azure Update Manager is responsible for keeping server systems up to date, and Windows Autopatch is a similar service for updating clients, which are typically not as closely integrated into the Azure infrastructure.

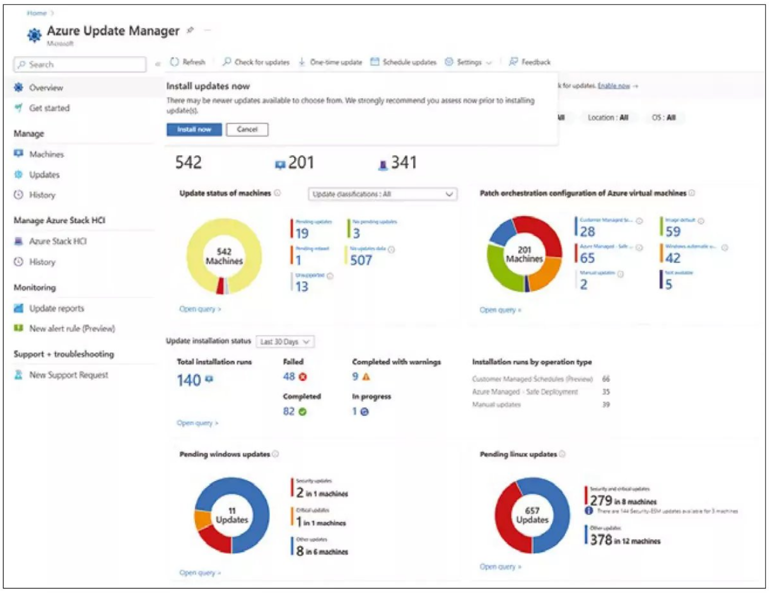
Two things should be noted: First, the full range of Autopatch features is only available if you have a Windows 10/11 Enterprise E3 or E5 license. If you use Microsoft 365, this is equivalent to license classes F3, E3, or E5. However, if you run your Windows clients on a business license, you will have to make do with a comparatively small subset of features, for which Microsoft offers a kind of menu [4].

Second, Windows Autopatch is essentially a cloud-based service. This setup leads to relatively high bandwidth requirements. If the Windows Delivery Optimization feature [5] is not set up correctly, each client will download its own update separately. In addition to the resulting periods of slow Internet connectivity, this situation can cause significant additional costs if Internet access is billed by volume.

In terms of its design, Windows Autopatch is an addition to the well-known Windows Intune, which provides numerous tools for endpoint management, including tools for controlling rollout processes.

Conclusions

The deprecation of WSUS has attracted some attention in the IT world. In practical terms, though, you do not need to rush into anything at this stage. It is impossible to predict what will happen in the fast-moving world of Microsoft between now and the end of the support cycle for Windows Server 2025. Of course, it can never hurt to take a look at successor services such as Azure Update Manager, Windows Autopatch, or third-party products (see the "Third-Party Providers Respond" box). ■


Home >
Azure Update Manager
Microsoft
Search
Refresh
Check for updates
One-time update
Schedule updates
Settings
Feedback
Overview
Get started
Manage
Machines
Updates
History
Manage Azure Stack HCI
Azure Stack HCI
History
Monitoring
Update reports
New alert rule (Preview)
Support + troubleshooting
New Support Request
Install updates now
Install now
Cancel
Enable now →
Location : All
OS : All
542
201
341
Update status of machines
Update classifications : All
542 Machines
Pending updates 19
No pending updates 3
Pending reboot 1
No updates data 507
Unsupported 13
Open query >
Patch orchestration configuration of Azure virtual machines
201 Machines
28
59
Azure Managed - Safe ... 65
Windows automatic ... 42
Manual updates 2
Not available 5
Open query >
Update installation status
Last 30 Days
Total installation runs 140
Failed 48
Completed with warnings 9
Completed 82
In progress 1
Open query >
Installation runs by operation type
Customer Managed Schedules (Preview) 66
Azure Managed - Safe Deployment 35
Manual updates 39
Pending windows updates
11 Updates
Security updates 2 in 1 machines
Critical updates 1 in 1 machines
Other updates 8 in 6 machines
Open query >
Pending linux updates
657 Updates
Security and critical updates 279 in 8 machines
Other updates 378 in 12 machines
Open query >

There may be newer updates available to choose from. We strongly recommend you assess now prior to installing update(s).

Figure 1: Azure Update Manager provides information about an update and the patch status.

Third-Party Providers Respond

Many large and small providers look to take on the patch management task. Besides tools such as deeploi [6], other platforms offer a range of functions reminiscent of WSUS. I cannot provide a complete overview of the tools available on the market in this article for reasons of space; however, it is interesting that some providers have already responded to the discontinuation of WSUS. One example is Action1 [7], a US start-up that promises completely free software patch management for users with up to 100 users or endpoints. As the end-of-life deadlines for WSUS approach, you can expect more providers to step into the breach.

Info

[1] Cost estimate for extended support: [https://blog.shi.com/business-of-it/weighing-the-costs-and-benefits-of-microsoft-extended-security-updates/]

[2] Support matrix for Azure Update Manager: [https://learn.microsoft.com/en-us/azure/update-manager/support-matrix?tabs=third-party-win]

[3] Microsoft Connected Cache: [https://learn.microsoft.com/en-us/windows/deployment/do/waas-microsoft-connected-cache]

[4] License requirements for Windows Autopatch: [https://learn.microsoft.com/en-us/windows/deployment/windows-autopatch/prepare/windows-autopatch-prerequisites]

[5] Delivery Optimization: [https://learn.microsoft.com/en-us/windows/deployment/do/waas-delivery-optimization]

[6] deeploi: [https://www.deeploi.io]

[7] Action1: [https://www.action1.com]

The Author

Tam Hanna (*tam.hanna* on Instagram) has seen the embedded space inside and out. His multidecade work has involved coding games for early mobile phones, designing metrology systems, and tackling various projects for civil and military clients.

Hone Your Skills
– with –
Special Issues!

Get to know Shell, LibreOffice, Linux, and more from our Special Issues library.

The *Linux Magazine* team has created a series of single volumes that give you a deep-dive into the topics you want.

Available in print or digital format

Check out the full library!
shop.linuxnewmedia.com/shop

BUILD A RASP PI RADIO!
MakerSpace #02
RASPBERRY PI GEEK
THE COMPLETE ARCHIVE
2,000 pages of maker projects and more!
FREE DVD $39.90 VALUE!
MakerSpace
HANDS-ON PROJECTS FOR MAKERS
RASP PI TRICKS
TRY ARTIFICIAL INTELLIGENCE ON A PI
MakerSpace #04
Retro Computing
Smart Home
Automate your house
Check plants and pets
Low-Code Programming
Learn Node-RED, then add Python
RASPBERRY PI GEEK
THE COMPLETE ARCHIVE
2,000 pages of maker projects and more!
DVD INSIDE $29.90 VALUE!
MakerSpace #05
HANDS-ON PROJECTS FOR MAKERS
Go Retro with DOS & BASIC
Proxmox for Pi
Run Virtual Machines on Your Raspberry Pi
RASP PI TRICKS
FEED YOUR FISH WHILE YOU'RE AWAY
MakerSpace #03
FIND THE PERFECT TOOL FOR THE TASK
2025 EDITION
Cool Linux Hacks
119 HACKS!
DISCOVER THE SECRETS OF EXPERTS
DVD INSIDE! LibreOffice Full Version
Dive deep into the world's greatest free office suite
2025/26 Edition
LibreOffice Expert
JOIN THE LINUX REVOLUTION!
ALL THE SOFTWARE YOU NEED!
FREE DVD!
openSUSE Leap 15.5
GETTING STARTED WITH LINUX
MORE POWERFUL • MORE SECURE • MORE FUN
THE BEST OF LINUX MAGAZINE 2024
Advanced Techniques from Linux Experts:
• Smoother operation
• Better security
• Smarter tools
Zint
Conjure up your own barcodes
Node-RED
Coding without the code
Shell Genie
Convert your natural-language questions to Linux commands
Photo Collages
Use these Linux tools to mash up your photos
ChatGPT Clients
AI at the Linux command line
Plus much more inside!
WWW.LINUX-MAGAZINE.COM
301 BEST BASH COMMANDS
LINUX SHELL HANDBOOK
2025 Edition
SUPERCHARGE YOUR LINUX SKILLS
Power at Your Fingertips
• Pipe and redirect output
• Monitor processes
• Create custom scripts
Keep this guide as a permanent reference!
INCLUDING INTERNET TOOLS
TERMINAL UTILITIES FOR EMAIL AND GOOGLE SEARCH
LINUX NEW MEDIA
WWW.LINUX-MAGAZINE.COM

Photo by Ricardo Gomez Angel on Unsplash

Find Entra ID vulnerabilities with AzureADRecon

Reconnaissance

The AzureADRecon tool lets you analyze Entra ID environments, generate reports, and identify potential risks at an early stage by providing detailed insights into users, roles, and service principals. We show you how to set up, automate, and use the tool for security checks. By Philip Lorenz

Countless organizations around the globe rely on Entra ID (formerly Azure AD) as their identity provider to organize authentication and access management for cloud and local applications, but it is precisely this widespread use that makes Entra ID an attractive target for attackers. The reason is clear: An attacker gaining access to identity management can potentially open the door to a wide range of protected applications and data, which is why cybercriminals use everything they have to gain initial access to Entra ID – whether through phishing, credential stuffing, or exploiting misconfigurations. Once on the inside, they focus on gathering information, identifying targets, and preparing further attacks.

Discovering Vulnerabilities

AzureADRecon [1] is a useful tool, enabling the systematic collection of information in the Entra ID environment, including user accounts, groups, roles, security policies, and configurations. However, you should note that the tool does not enable unauthorized access; rather, it is used to extract information quickly once initial access to the environment has been made.

The tool offers added value for administrators and security officers: They can view their own Entra ID from the perspective of a potential attacker, detect vulnerabilities at an early stage, and strengthen security measures in a targeted way. The benefit is a quick and easy way to scan your infrastructure before third parties do.

Anyone who thinks they need to dive deep into PowerShell to take a look behind the scenes of Azure Entra ID is mistaken. Armed with AzureADRecon, gaining in-depth insights into the environment is a matter of a few simple steps. The tool is intuitive to use and provides valuable information for IT administrators and security managers alike at the push of a button.

Entra ID as a Control Center

Entra ID is Microsoft's cloud-based identity and access management solution. It acts as the central building block for authentication and authorization in hybrid and cloud-only IT infrastructures. Entra ID enables users to access applications, services, and resources, whether in the cloud, locally, or in hybrid scenarios. This comprehensive functionality makes it an indispensable tool for businesses of all sizes.

As an integral part of Microsoft's ecosystem, Entra ID is frequently used for (1) single sign-on (SSO), which enables employees to log in seamlessly to different applications with a single set of credentials, improving the user experience while minimizing security risks; (2) multifactor authentication (MFA), which prevents unauthorized access to sensitive resources; (3) identity management, which provides central management functions for user accounts, roles, and access authorizations; and (4) integrated security policies, with features such as conditional access to help enforce security policies dynamically on the basis of user location, device, or behavior.

Popular Target for Attackers

Entra ID is an attractive target for cybercriminals because it acts as a gatekeeper for cloud resources: Once attackers gain privileged access (e.g., administrator authorizations) to an Entra ID account, they can access a wide range of sensitive organizational resources, including:

1. Access to business-critical applications: Entra ID is linked to important services such as M365, Exchange Online, SharePoint, Teams, and OneDrive. A compromised user account can give attackers immediate access to these business-critical applications, which means that criminals can steal company data, sniff communications, or misuse email for phishing campaigns.
2. Access to privileged accounts: If attackers hijack a global admin account or an account with elevated privileges, they have access to administrative functions, which means they can create new users, reset passwords, change access controls, or even create backdoor accounts to secure long-term access to the environment.
3. Directory enumeration and reconnaissance: Entra ID provides access to user and group information, which attackers who have access to a user account can use to identify the users, groups, and service principals that exist in the environment. This knowledge can be used as a basis for further attacks, such as phishing, social engineering, or lateral movement.

Getting Started with the Cyber Kill Chain

The cyber kill chain describes the typical phases of a cyberattack – from initial reconnaissance to complete exploitation of the target environment (**Figure 1**). In the reconnaissance phase, the first step in the kill chain, attackers gather specific information about their target environment. The goal is to identify vulnerabilities and plan the next steps, which is exactly where AzureADRecon comes into play – the name says it all.

After gaining initial access to the Entra ID environment, cybercriminals use AzureADRecon to extract internal information systematically, including user accounts, groups, roles, permissions, and registered applications. The data reveals which accounts are particularly sensitive and where potential attack vectors for privilege escalation might be hidden. With its automated data collection feature, AzureADRecon offers an efficient way to scan the entire Entra ID environment with just a few commands. For attackers, this crucial step paves the way for the next phases of the kill chain, such as privilege escalation and lateral movement.

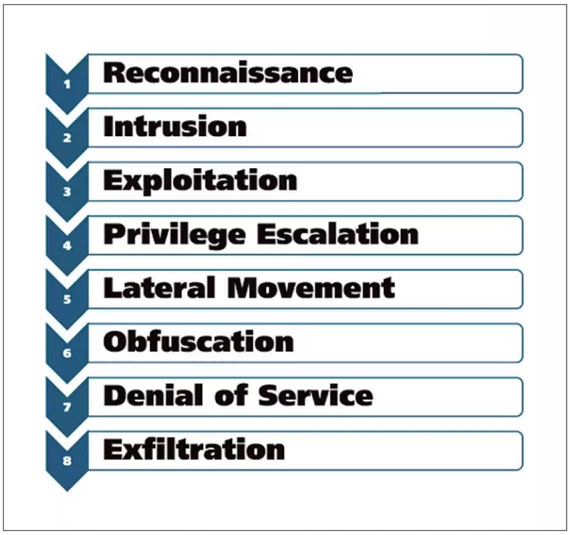


Figure 1: The cyber kill chain is a model that describes the lifecycle of a cyberattack. AzureADRecon starts with the recon(naissance) phase.

Overview and Installation

The open source AzureADRecon systematically collects detailed information from an Entra ID environment and presents it in a clear-cut format. You can export the results in various formats, including HTML and Excel, which makes evaluation and downstream processing far easier. With just a few commands, AzureADRecon reads specific information and summarizes it in a structured report (**Figure 2**), with information on Tenants, Domains, Licenses, Users, Service Principals, DirectoryRoles, DirectoryRoleMembers, Groups, GroupMembers, and Devices.

Thanks to automated collection, AzureADRecon provides a quick approach to achieving a comprehensive overview of the Entra ID environment. The tool is flexible and can be used for security analyses or to optimize your own IT environment.

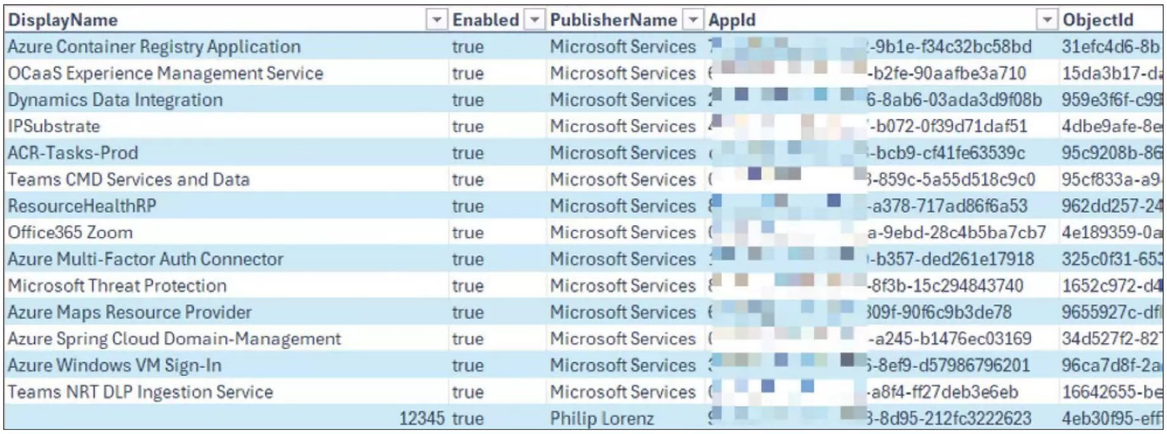
Installation and Dependencies

The requirements listed in the box are essential for AzureADRecon to work smoothly. The tool is not a PowerShell module from PSGallery; instead, you get the code directly from GitHub in one of three ways: either by downloading as a ZIP file from the GitHub web interface [2], by cloning the Git repository,

```
git clone https://github.com/➘
        adrecon/AzureADRecon.git
```

or by downloading my PowerShell one-liner:

```
iex (➘
  iwr "https://raw.githubusercontent.com/➘
      philiplorenz-code/AzureADRecon/➘
      refs/heads/feature/setup/➘
      install.ps1"
  -UseBasicParsing)
```



DisplayName	Enabled	PublisherName	AppId	ObjectId
Azure Container Registry Application	true	Microsoft Services	[redacted]-9b1e-f34c32bc58bd	31efc4d6-8b
OCaaS Experience Management Service	true	Microsoft Services	[redacted]-b2fe-90aafbe3a710	15da3b17-d
Dynamics Data Integration	true	Microsoft Services	[redacted]6-8ab6-03ada3d9f08b	959e3f6f-c9
IPSubstrate	true	Microsoft Services	[redacted]-b072-0f39d71daf51	4dbe9afe-8
ACR-Tasks-Prod	true	Microsoft Services	[redacted]-bcb9-cf41fe63539c	95c9208b-8
Teams CMD Services and Data	true	Microsoft Services	[redacted]-859c-5a55d518c9c0	95cf833a-a9
ResourceHealthRP	true	Microsoft Services	[redacted]-a378-717ad86f6a53	962dd257-2
Office365 Zoom	true	Microsoft Services	[redacted]a-9ebd-28c4b5ba7cb7	4e189359-0
Azure Multi-Factor Auth Connector	true	Microsoft Services	[redacted]-b357-ded261e17918	325c0f31-65
Microsoft Threat Protection	true	Microsoft Services	[redacted]-8f3b-15c294843740	1652c972-d
Azure Maps Resource Provider	true	Microsoft Services	[redacted]09f-90f6c9b3de78	9655927c-d
Azure Spring Cloud Domain-Management	true	Microsoft Services	[redacted]-a245-b1476ec03169	34d527f2-82
Azure Windows VM Sign-In	true	Microsoft Services	[redacted]-8ef9-d57986796201	96ca7d8f-2
Teams NRT DLP Ingestion Service	true	Microsoft Services	[redacted]-a8f4-ff27deb3e6eb	16642655-b
12345	true	Philip Lorenz	[redacted]-8d95-212fc3222623	4eb30f95-ef

Figure 2: The service principals collected by AzureADRecon list apps that use Entra ID as their identity provider, providing clues as to which services could potentially be exploited if accounts are compromised.

The script also installs the necessary prerequisites (see the "AzureADRecon Prerequisites" box).

Generating Reports

To start AzureADRecon, open a PowerShell console and enter the path to the script (`C:\Scripts\ AzureADRecon.ps1`). Press Enter to confirm, and, if you downloaded the script with the single-line installation, run it immediately.

When launched, the tool prompts you to authenticate against Entra ID. The environment scan then begins. The generated CSV files and the aggregated Excel report are saved in the specified output directory. The Excel report provides a clear overview of the Entra ID objects, including user accounts, properties, and stored licenses.

The acquired service principals are particularly interesting for attackers. They provide insights into apps that use Entra ID as an identity provider and indicate which services could potentially be exploited if accounts are compromised. This report is also valuable for administrators to track down app registrations that are no longer in use.

AzureADRecon offers the option of passing in various parameters when starting the script. You can specify the parameters directly; for example:

```
C:\Scripts\AzureADRecon.ps1 ➘
  -OutputType "Excel" ➘
  -Log
```

An overview of the most important parameters is given in Table 1. These parameters give you more granular control over the scan and report output, which is ideal for tailoring the results to specific requirements.

Automated Scans

To receive regular updates from the Entra ID environment, I recommend running AzureADRecon automatically with Windows Task Scheduler

AzureADRecon Prerequisites

- .NET Framework 3.0 or newer (included in Windows 7 and later)
- PowerShell 2.0 or newer (included in Windows 7 and later)
- AzureAD PowerShell Module 3.0 or newer – installable with `Install-Module -Name AzureAD`
- Microsoft Graph Module for PowerShell Core (Excel generation requires Windows) and the permissions (scopes) `AuditLog.Read.All`, `User.Read.All`, and `UserAuthenticationMethod.Read.All` – installed with `Install-Module -Name Microsoft.Graph`
- Microsoft Excel (recommended for report generation)

Table 1: Important AzureADRecon Parameters

Parameter	Description	Default
`-Method`	Method to be used: `AzureAD` or `MSGraph`	`AzureAD` unless a non-Windows host
`-Credential`	Credentials for the domain	–
`-GenExcel`	Generates the aggregated Excel file in the output folder	–
`-TenantID`	Tenant ID for multitenant environments	–
`-OutputDir`	Storage path for the output files	`pwd`
`-Collect`	Modules to be collected (comma separated)	All modules
`-OutputType`	Output type: `CSV`, `Excel`, `JSON`, `HTML`	STDOUT with `-Collect` parameter, else CSV and Excel
`-DormantTimeSpan`	Period for inactive accounts (days)	`30`
`-PassMaxAge`	Maximum password age for machine accounts (days)	`90`
`-Threads`	Number of threads for processing	`10`
`-Log`	Enables the creation of a log (start transcript)	–

(e.g., to run the script at 3am every night).

Regular automation provides daily updated reports without having to run the tool manually, enabling proactive monitoring of the environment and helping you identify potential vulnerabilities at an early stage.

Scheduled tasks can of course be created in the GUI. Alternatively, you can use the PowerShell code in Listing 1. All you need to do is adjust the values to suit your needs. The most important variable is `$scriptPath`, which specifies which script you want to run regularly. For the task to work properly, you need the appropriate PowerShell script that handles the AzureADRecon call and provides the required parameters (Listing 2). Save this script in the `runazadrecon.ps1` file.

Secure Authentication Without MFA

The script requires login credentials (username and password) to enable automated login. Because MFA prevents automated access, a user that does not require MFA must be used here. I strongly recommend that you do not store the password directly in the script – but store it securely. A common approach is to use the PowerShell secret vault. For simplicity, I used the password directly in the script in this example.

To test whether the scheduled task works as expected, open the GUI, right-click on the task, and select *Run*. The data should then be collected in Excel format in the directory defined under `$outputDir`.

Conclusions

AzureADRecon proves how efficiently targeted information can be collected from an Entra ID environment with minimal overhead. The tool is not just valuable for administrators and security managers, in that it helps to identify vulnerabilities at an early stage – it also sharpens the eye for the perspective of potential attackers. Its ease of use and versatile export options, especially the aggregated Excel report, make AzureADRecon a very useful tool that you can use to track down unused resources, check configurations, and proactively strengthen security. ■

Info

[1] **AzureADRecon in GitHub:** [https://github.com/adrecon/AzureADRecon]

[2] **AzureADRecon as ZIP download:** [https://github.com/adrecon/AzureADRecon/archive/master.zip]

Listing 1: Creating a Scheduled Task

```
$taskName = "RunAzADRecon" # Name of the scheduled task
$taskDescription = "Runs AzureADRecon at 3 a.m. every night"
$scriptPath = "C:\code\runazadrecon.ps1" # Path to the PowerShell script to be executed
# Define triggers that execute the task
$trigger = New-ScheduledTaskTrigger -Daily -At "03:00AM"
# Define the action that the task should perform
$action = New-ScheduledTaskAction -Execute "powershell.exe" -Argument "-NoProfile -WindowStyle Hidden
                                  -File $scriptPath"
# Optional: Define task settings
$settings = New-ScheduledTaskSettingsSet -AllowStartIfOnBatteries -DontStopIfGoingOnBatteries
                                         -StartWhenAvailable
# Register new task
$registerParameters = @{TaskName = $taskParameters.TaskName Description = $taskParameters.Description
                        Trigger = $trigger Action = $action Settings = $settings}
Register-ScheduledTask @registerParameters
```

Listing 2: runazadrecon.ps1

```
# Store login details
$mail = "<User>"
$pass = "<Password>"
$outputDir = "<Path>" # Location where the reports are saved
$parameters = @{Credential = New-Object System.Management.Automation.PSCredential
                ($mail,(ConvertTo-SecureString $pass -AsPlainText -Force))OutputDir = $outputDir|}
# Call the script with splatting
C:\Scripts\AzureADRecon.ps1 @parameters
```

Lead Image © dunord, 123RF.com

Proactive threat detection

Silent Observer

The Falco tool promises comprehensive threat detection for Docker, Podman, and Kubernetes from network traffic streams. By Martin Loschwitz

Regular checks and comprehensive compliance are no longer enough to guarantee security in data centers. Because criminals are getting more savvy and their attacks are becoming increasingly complex and sophisticated, defenses need to keep pace. Proactive monitoring of network traffic has been the order of the day for some time now. Partly in the wake of AI, this approach involves identifying unusual activity from suspicious data packets in network traffic and taking preventive measures before an attacker can exploit the opportunity.

Many services of this kind are on the market, Coroot [1] being just one that has made a name for itself (Figure 1). However, this tool is currently facing some reservations, particularly among European

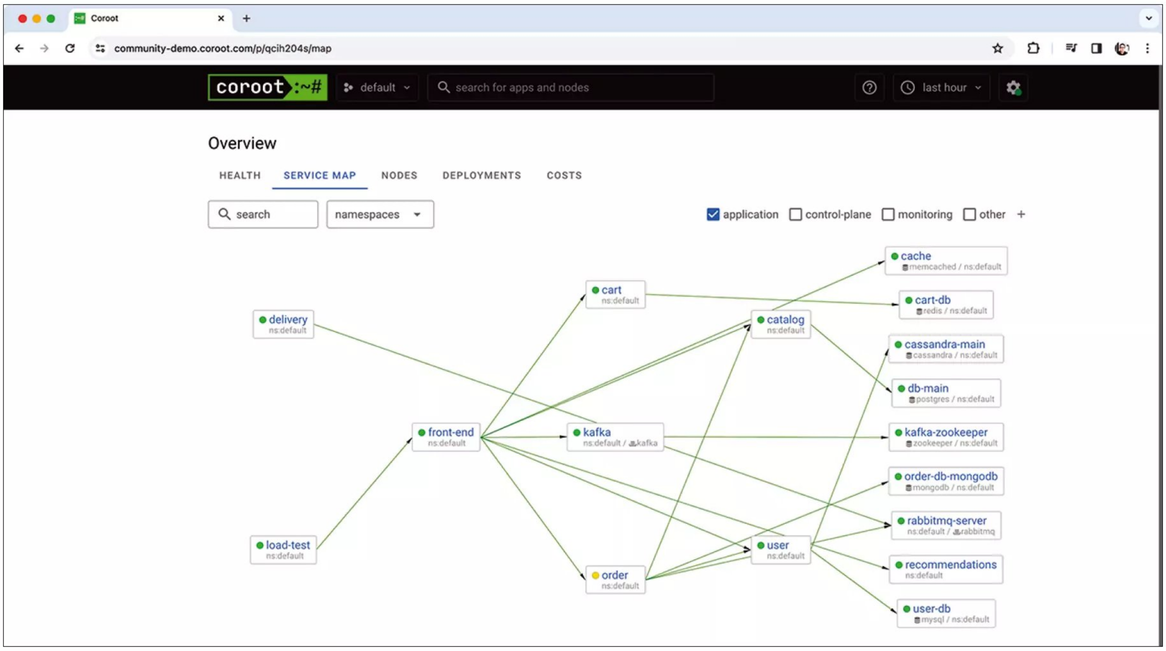

Coroot
community-demo.coroot.com/p/qcih204s/map
coroot :~#
default
search for apps and nodes
last hour
Overview
HEALTH
SERVICE MAP
NODES
DEPLOYMENTS
COSTS
search
namespaces
☑ application
☐ control-plane
☐ monitoring
☐ other
+
cache memcached / ns:default
cart ns:default
cart-db redis / ns:default
delivery ns:default
catalog ns:default
cassandra-main cassandra / ns:default
db-main postgres / ns:default
front-end ns:default
kafka ns:default / kafka
kafka-zookeeper zookeeper / ns:default
order-db-mongodb mongodb / ns:default
rabbitmq-server ns:default / rabbitmq
load-test ns:default
user ns:default
recommendations ns:default
order ns:default
user-db mysql / ns:default

Figure 1: Coroot performs security analyses with AI but uploads a large volume of data to the cloud for this purpose, whereas Falco is exclusively local in this respect.

customers, because it can only be used if you are prepared to upload comprehensive information about your own data traffic to the cloud, where the AI part of the application is located.

A far more down-to-earth solution by the name of Falco [2] aims to impress with the use of less AI. Its creators describe their tool as cloud-native threat detection, optimally integrated with Kubernetes and classic runtime environments such as Docker or Podman. Manufacturer Sysdig has invested a huge amount of time and effort in Falco. In 2024, after the typical incubation phase, it became an official project of the Cloud Native Computing Foundation (CNCF), which is reason enough to take a closer look at how, and how well, active threat detection with Falco works.

Container Challenges

Containers have become indispensible tools in the Linux world. Major vendors such as Red Hat and SUSE have long since begun to convert their enterprise distributions, once considered their crown jewels, into container players. From the vendor's point of view, it has the massive advantage of allowing them to leave the maintenance of the application ecosystem to third-party providers of the respective tools. However, in the security context, container-based workloads come with challenges that do not exist in conventional application scenarios, primarily related to security during execution (i.e., at runtime). This scenario is exactly where Falco steps in.

The challenge – and a key selling point for containers – is that workloads are (intentionally) isolated from the rest of the system by the runtime environment. Containers use their own libraries and a virtualized network connection. They can only access the host's filesystem in strictly defined ways. The downside of this isolation is that typical threat detection tools are unable to see much of what the containers and the applications running inside them are doing.

Attack vectors are many; for example, the libraries and tools in containers are not normally covered by the host's software manager, which means that, in container-based environments, you cannot just rely on updating packages once a quarter. Although you are updating the host's software and the runtime environment for the containers (e.g., Podman or Docker), the software in the containers does not benefit.

On an Ubuntu system running a containerized Ceph, for example, you have to deal with both CentOS Stream and Ubuntu, because CentOS Stream is the distribution that Red Hat uses for its Ceph containers. If other containerized applications are then added that use different base distributions, the result is complete chaos. From the system administrator's point of view, this can only be managed by continuously updating the containers. If you don't want to have to check all your systems personally every day for zero-day vulnerabilities, then you need software that independently detects any threats and raises the alert, and Falco promises to do precisely these tasks (Figure 2).

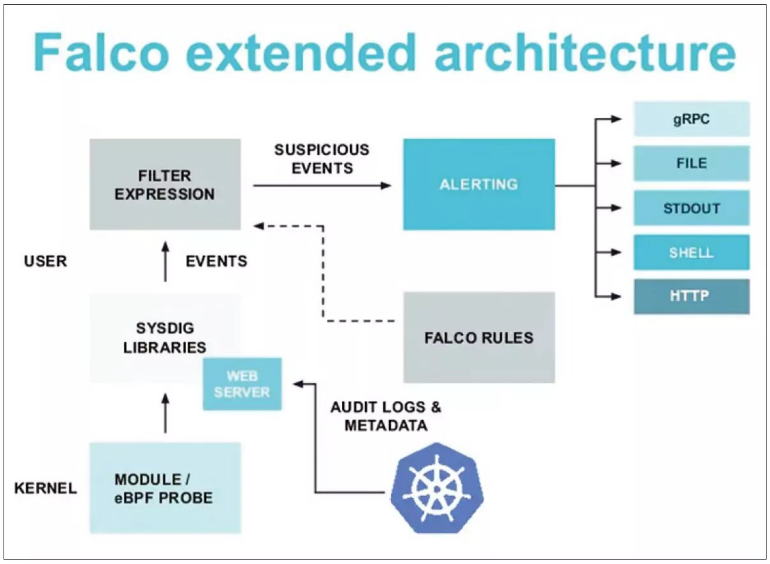


Figure 2: Falco essentially comprises a ruleset and plugins for data sources such as eBPF. In terms of output, it supports various services. © Falco project

Even More Problems

Faulty applications in the container itself are only one problem area. Other scenarios are now also widespread. Security vulnerabilities (in the worst case, unknown ones) in the runtime environment could help attackers break out of the container and gain access to the host's resources. This kind of privilege escalation normally leaves clear traces, but a Linux system with a default configuration and Docker CE does not have the tools to detect it automatically, which is why Falco also covers this vector.

Finally, faulty software in containers can become a problem, such as in situations in which the active software does not manage its memory correctly and, in particular, does not release it thoroughly enough when no longer needed. If attackers encounter a faulty application of this kind, they can often crash the application by skillfully bombarding it with requests, indirectly crashing other containers on the same host. In the worst case, they could even take down the entire environment.

For this reason, Falco also comes with functions that monitor application behavior and detect errors such as memory leaks without a need for administrator intervention. Importantly, Falco does not focus solely on individual containers and runtime environments on individual systems. Instead, Falco can be fully integrated into Kubernetes, today's most popular container orchestrator. This talent makes it easy for developers and system administrators to have Falco monitor applications on their systems.

Falco Architecture

To understand Falco, you should look at its architecture. The manufacturer originally developed the tool as an integral part of another piece of software, but today Falco is an excellent standalone tool that comprises several components. From the user's point of view, the most important piece of software is the `falco` command-line program. With this tool, you can configure the software comprehensively. You just need to specify the events for which you want Falco to respond and the kind of response you want in each case.

The key Falco component is the driver. The Falco developers use the term "driver" to describe a program that is directly connected to the Linux kernel and generates a data stream from the events in the containerized applications, which it then forwards to Falco. The driver can currently come in several forms, with an extended Berkeley Packet Filter (eBPF) probe being the standard. Falco still includes an older and less powerful version for reasons of downward compatibility. A kernel module for Falco generates an event stream that allows it to process after loading, which makes it clear that Falco is not an entirely new solution, because the kernel driver is, strictly speaking, legacy software and dates back to the days when eBPF [3] was not available to the extent it is today.

Importantly, for a driver to feed Falco meaningful information, the service needs a definition of its event sources: specifically, informing the kernel driver or eBPF which events you want to be taken into account and to which notifications the respective driver needs to pay attention.

Falco is very versatile in this respect, because event sources can be defined dynamically in Falco or removed by a plugin mechanism [4].

Finally, communication with the outside world is also important. The software offers the option of connecting to various external services to notify you directly of specific events.

eBPF: The Game Changer

As mentioned earlier, Falco's preferred mechanism for retrieving data is eBPF (Figure 3). Because this tool plays a central role in Falco, a brief detour through the eBPF camp is useful. Although the packet filter extension has been an integral part of Linux for some time, not many tools yet use eBPF as efficiently as Falco does. EBPF is the original packet filter of the Berkeley Software Distribution (BSD), a Unix-like operating system. Experts have loved this extremely powerful and versatile filter for decades, which is why it was ported to Linux.

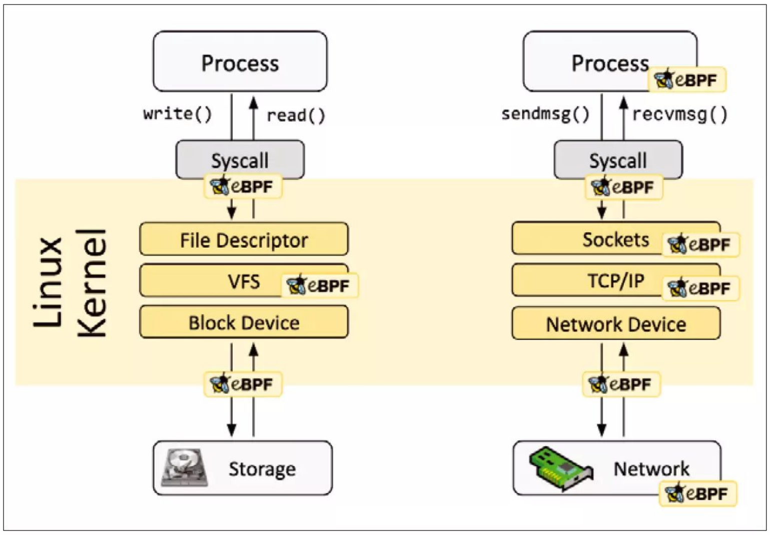


Figure 3: Because of its fast performance, eBPF is considered the gold standard in Linux when it comes to accurate analysis of traffic. © eBPF project

Linux has no shortage of tools for packet filtering, including Netfilter and its various front ends (e.g., iptables and, more recently, nftables).

To say that eBPF is "extended" is an understatement. The technology is nothing less than a virtualization environment. eBPF uses a special form of virtualization whose goal is to execute arbitrary programs, known as probes, in the Linux kernel's network stack.

EBPF can record data in transit within a Linux system and modify the data as desired. One key argument put forward by eBPF advocates is its extremely fast performance. Today's data centers are no longer looking at bandwidths in the 1Gbps range; instead, 25Gbps is the norm, and even 400Gbps is no longer uncommon.

Other packet filters for Linux, most notably the now obsolete iptables, are simply too slow. However, eBPF provides developers and administrators with an interface in the Linux kernel that allows them to examine and manipulate packets in real time, even at high bandwidths.

The Falco developers primarily focus on analyzing active data streams.

One particularly practical feature, unlike other packet filters, is that eBPF probes let you use the traffic, including sources and targets, as parameters. Data connections between individual microarchitecture applications of an app, for example, can be analyzed in a far better way with eBPF than with other solutions, because an eBPF probe also precisely identifies the receiving and sending programs on the system. It is this feature that makes the combination of Falco and eBPF so effective.

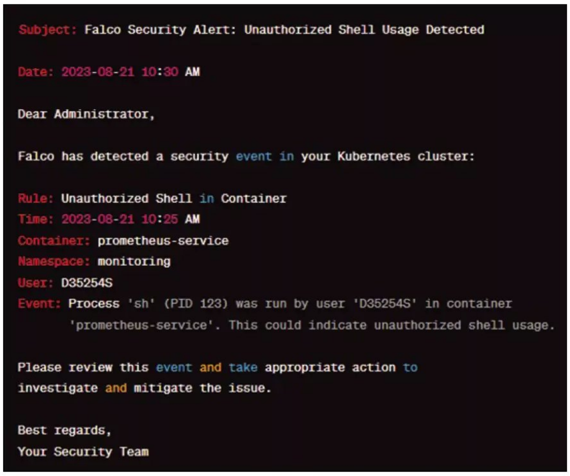
Accurate Logger

Essentially, Falco simply uses eBPF to tap into a system's network traffic and probe the traffic there as a function of the stored configuration. The Falco kernel module mentioned earlier dates back to a time when eBPF did not yet exist, or when the technology was not yet mature enough for production, which explains why the module is only found in legacy installations today.

Newer Falco setups should always use the eBPF source for Falco. Besides eBPF, Falco supports other methods for analyzing application behavior. For example, when acting as a sidecar to an app, the tool can also examine and evaluate the app's syscalls, which is particularly important where threat analysis not only relates to network traffic but also the general behavior of applications – for example, with regard to their memory usage.

Filtering by Rules

To find out how Falco analyzes traffic, look behind the scenes to find the Falco rules engine [5] at work. The engine is where the instructions that determine Falco's behavior are stored in a native, declarative syntax. A Falco rule comprises at least a unique name, a description, a list of conditions that must be met for the rule to take effect, and a list of outputs that the rule generates. Moreover, the priority that applies to events that match a specific rule in Falco must be defined.



```
Subject: Falco Security Alert: Unauthorized Shell Usage Detected

Date: 2023-08-21 10:30 AM

Dear Administrator,

Falco has detected a security event in your Kubernetes cluster:

Rule: Unauthorized Shell in Container
Time: 2023-08-21 10:25 AM
Container: prometheus-service
Namespace: monitoring
User: D35254S
Event: Process 'sh' (PID 123) was run by user 'D35254S' in container
       'prometheus-service'. This could indicate unauthorized shell usage.

Please review this event and take appropriate action to
investigate and mitigate the issue.

Best regards,
Your Security Team
```

Figure 4: Falco uses system calls and analyzes network traffic to detect potentially problematic events, such as shell calls inside a container.

From an admin's perspective, the fields for conditions, outputs [6], and priority are of particular interest. Listing 1 shows an example taken directly from the Falco documentation and clearly illustrates what the tool is capable of doing with relatively little effort.

Listing 1: Example Rule

```
rule: shell_in_container
desc: notice shell activity within a container
condition: >
  evt.type = execve and
  evt.dir = < and
  container.id != host and
  (proc.name = bash or
  proc.name = ksh)
output: >
shell in a container | user=%user.name container_
  id=%container.id container_name=%container.name
shell=%proc.name parent=%proc.pname
cmdline=%proc.cmdline
priority: WARNING
```

In this example, `rule` defines the rule name and `desc` supplies a description of the rule's purpose. Of course, the list of conditions is far more interesting. It is easy to see that several conditions are combined here, each connected by an `and` statement. For the rule to take effect, all conditions must be met. The `evt.type` specifies the type of event; in this example it means running the `execve` system call. The `evt.dir` line describes the communication direction, where > specifies the beginning of a call and < its end. The `container.id` is pretty much what it says on the label; in this example it states that the ID of the container must not be identical to the host's imaginary ID.

In this specific example, the rule for Falco is designed to identify and report shells being called in containers, but not on the host. Accordingly, the `proc.name` restrictions, which specify the name of the invoked program, apply here. Finally, the `output` specification defines the log message that Falco outputs for an event meeting all the criteria specified in `condition`. The `priority` argument specifies the warning level, which makes clear the purpose of the example rule in Listing 1: It outputs a warning every time a shell in the form of Bash or Ksh is called in a container running on the host (Figure 4).

Working with Warnings

Of course, a warning alone is not very useful if you have no way to forward it directly to the admin. Falco offers a plethora of options for creating warnings, but how can you ensure that an alert is sent? This is where the Falco output plugins enter the scene. They offer a wide range of options for sending warnings from Falco directly to external services.

Besides popular chat platforms such as Slack [7] and Teams [8], the tool supports classic instant messaging services. Falco can also directly address various pager services such as PagerDuty [9] and Opsgenie [10]. A REST interface is available for connecting to services that deliver short messages. Falco is also compatible with various monitoring systems, including Prometheus Alert Manager [11], Nagios [12], and Icinga [13], with the advantage that existing alert chains and escalation levels can be rehashed without any problems. Once Falco is connected to the existing monitoring system, alerts are triggered automatically.

Teamed with Kubernetes

Falco is practical, but, in most cases, it is unlikely to be used exclusively in combination with a container runtime environment such as Docker or Podman. Instead, you will primarily want to use Falco in combination with Kubernetes, the software of choice when it comes to orchestrating fleets of containers.

The good news is that the integration of Falco and Kubernetes (K8s) has largely been completed. For example, the project itself offers Helm charts (i.e., metadata for installing Falco in Kubernetes-based clusters with the K8s package manager Helm). Alternatively, an operator is available that launches Falco in Kubernetes. What both approaches have in common is that they let you start and manage Falco instances and that Falco can be made available as a sidecar in one or multiple containers. Falco then examines the processes in the rolled-out pods according to the defined rules and analyzes the network traffic as specified in the configuration.

Even More Input

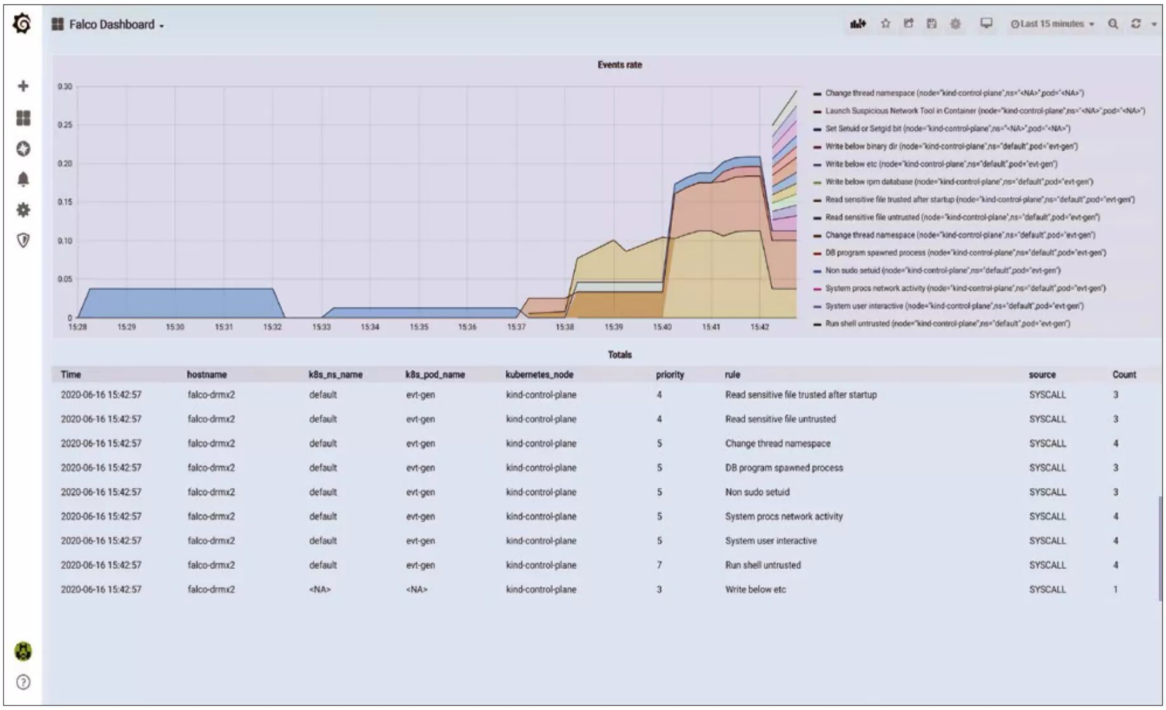
In a standard deployment scenario, Falco primarily receives data for analysis from eBPF or the kernel module, as described previously. However, Falco's plugin architecture is so generic that it can also be used to field event messages from other services and evaluate them according to the existing ruleset. For example, Kubernetes is regularly used in conjunction with single sign-on (SSO) services, such as Okta, which can produce and output their own log messages (Figure 5).

To remove the need to configure each component of the environment separately to field the data from potential alerts, you will find plugins that let you connect certain external services to Falco. All told, these help Falco create genuinely useful audit trails for events in Kubernetes.

Falco can also be linked to the Kubernetes audit component, `k8saudit`. Audit-releated events then end up as messages in Falco and can serve as the basis for alerts. Moreover, the Falco developer team is actively working on new plugins. For example, a module is planned that will allow an AI model for adaptive threat prediction to be developed and adapted to local conditions. After appropriate training, the model should be able to detect anomalies automatically in network traffic and therefore actively prevent attacks.

What Falco Cannot Do (Right Now)

Even during your first steps with Falco, you will quickly see that it is a very powerful tool, so it is important to understand what Falco is and what it is not. Many administrators would intuitively expect a runtime security tool to have a built-in scanner for vulnerable software. In other words, they would like a tool that logs in to running containers, examines the version numbers of the installed software, and fires off an alert if a CVE exists that documents a vulnerability in the given version. On the one hand, Falco does have the ability to detect attacks. For example, if a shell is suddenly executed


Falco Dashboard
Last 15 minutes
Events rate
0.30
0.25
0.20
0.15
0.10
0.05
0
15:28
15:29
15:30
15:31
15:32
15:33
15:34
15:35
15:36
15:37
15:38
15:39
15:40
15:41
15:42
Change thread namespace (node="kind-control-plane",ns="<NA>",pod="<NA>")
Launch Suspicious Network Tool in Container (node="kind-control-plane",ns="<NA>",pod="<NA>")
Set Setuid or Setgid bit (node="kind-control-plane",ns="<NA>",pod="<NA>")
Write below binary dir (node="kind-control-plane",ns="default",pod="evt-gen")
Write below etc (node="kind-control-plane",ns="default",pod="evt-gen")
Write below rpm database (node="kind-control-plane",ns="default",pod="evt-gen")
Read sensitive file trusted after startup (node="kind-control-plane",ns="default",pod="evt-gen")
Read sensitive file untrusted (node="kind-control-plane",ns="default",pod="evt-gen")
Change thread namespace (node="kind-control-plane",ns="default",pod="evt-gen")
DB program spawned process (node="kind-control-plane",ns="default",pod="evt-gen")
Non sudo setuid (node="kind-control-plane",ns="default",pod="evt-gen")
System procs network activity (node="kind-control-plane",ns="default",pod="evt-gen")
System user interactive (node="kind-control-plane",ns="default",pod="evt-gen")
Run shell untrusted (node="kind-control-plane",ns="default",pod="evt-gen")
Totals

Time	hostname	k8s_ns_name	k8s_pod_name	kubernetes_node	priority	rule	source	Count
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	4	Read sensitive file trusted after startup	SYSCALL	3
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	4	Read sensitive file untrusted	SYSCALL	3
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	5	Change thread namespace	SYSCALL	4
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	5	DB program spawned process	SYSCALL	3
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	5	Non sudo setuid	SYSCALL	3
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	5	System procs network activity	SYSCALL	4
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	5	System user interactive	SYSCALL	4
2020-06-16 15:42:57	falco-drmx2	default	evt-gen	kind-control-plane	7	Run shell untrusted	SYSCALL	4
2020-06-16 15:42:57	falco-drmx2	<NA>	<NA>	kind-control-plane	3	Write below etc	SYSCALL	1

Figure 5: Outputs can be used to transfer data from Falco to other services such as Prometheus, where the data can be processed and analyzed. © Falco project

in a container, it is always a cause for concern, but Falco detects it without any problems, as shown in the example in **Listing 1**. On the other hand, though, version-based attack detection does not work here. Instead, Falco actively scans the container's network traffic for signs of intrusions and raises the alarm if necessary. The philosophy is clear: With so many vulnerability scanners in many different forms for containers, the Falco developers see no reason to re-invent the wheel. Instead, Falco detects problems in other ways.

By the way, you do not necessarily have to create Falco rulesets laboriously from scratch yourself. The community already has several Git directories with ready-made rules for many areas of application that can be easily transferred to an existing Falco setup and enabled there.

Conclusions

Falco is a very powerful tool for threat detection and damage mitigation in environments where containers are the rule of the day. The tool really comes into its own when combined with Kubernetes, integrating seamlessly with K8s and monitoring the containers running in it on the basis of their communication, without a need to adapt the application. Falco differs from other monitoring and alerting solutions then, such as those offered by the OpenTelemetry project [14]. Where other solutions require a service to be prepared specifically for use with OpenTelemetry and compiled against the OpenTelemetry tools, Falco can be used with any tool in any container running on a standard Linux system. The program's quality is impressive, as is the large community working on Falco. At the end of the day, you don't have too many alternatives, either. The only obvious one would be Tetragon [15], but if you're dealing with runtime threat detection in containers, you will definitely want to take a closer look at Falco.

Info

[1] Coroot: [https://coroot.com/]
[2] Falco: [https://github.com/falcosecurity/falco]
[3] eBPF: [https://ebpf.io/]
[4] Falco plugins: [https://github.com/falcosecurity/plugins]
[5] Falco rules: [https://github.com/falcosecurity/rules]
[6] Falco output tools: [https://falco.org/docs/concepts/outputs]
[7] Slack: [https://slack.com]
[8] MS Teams: [https://www.microsoft.com/en-us/microsoft-teams/group-chat-software]
[9] PagerDuty: [https://www.pagerduty.com]
[10] Opsgenie (discontinued): [https://www.atlassian.com/software/opsgenie/migration]
[11] Prometheus: [https://www.cncf.io/projects/prometheus]
[12] Nagios: [https://www.nagios.org]
[13] Icinga: [https://icinga.com]
[14] Open Telemetry: [https://opentelemetry.io]
[15] Tetragon: [https://tetragon.io]

The Author

Martin Loschwitz is the founder and managing director of True West IT Services GmbH, which offers scalable IT infrastructure based on OpenStack and Kubernetes.

Microsoft Active Directory Rights Management Services

Know Your Rights

A built-in Microsoft AD RMS server role can help prevent data loss in your organization. By Conor Fitzgerald

One of the most dreaded calls to be received by an IT administrator is that a former employee is suspected of stealing intellectual property or financial or personal data. Your task is now to discover the extent of the data loss. Although many large organizations benefit from the data loss prevention (DLP) technologies built into modern cloud offerings such as Microsoft 365 E5 licencing and Azure Rights Management, it is fair to say that not all organizations have the budget to implement such solutions, leaving them exposed to the real risks of data loss. Implementing DLP strategies and solutions can be extremely expensive, but Microsoft has some built-in tools that could be deployed to help alleviate the risks of data loss, even if you have a limited budget. One such tool is Microsoft Active Directory Rights Management Services (AD RMS).

Microsoft AD RMS

Microsoft AD RMS is an on-premises information protection technology that helps organizations protect data from unauthorized access. AD RMS was first introduced in Microsoft Windows Server 2003 and is available as a built-in role in all major versions of Windows Server operating systems, up to and including Windows Server 2025. Although AD RMS is considered a legacy technology compared with modern solutions such as Azure Rights Management, it is fair to say that AD RMS can still play a major part in any DLP strategy, especially if the majority of your data is still hosted on in-house file servers and file shares.

As the name suggests, AD RMS requires an Active Directory domain environment, and the server running the AD RMS role needs to be domain joined. AD RMS should not be installed on a domain controller, and the recommended best practice would be to set up a dedicated server cluster for it, with a minimum of two AD RMS servers for resiliency and high availability. Installing the AD RMS role (Figure 1) will also automatically install required features such as:

- Internet Information Services (IIS) web server
- Windows Process Activation Service (WAS)
- Microsoft Message Queuing (MSMQ)
- .NET Framework

Other requirements include having access to a Microsoft SQL Server or SQL Server Express instance for configuration and logging databases, along with a dedicated service account in your Active Directory domain and an SSL certificate. The certificate can be self-signed if you do not have access to a certificate authority, but ideally it should be a certificate generated by your public key infrastructure (PKI) environment.

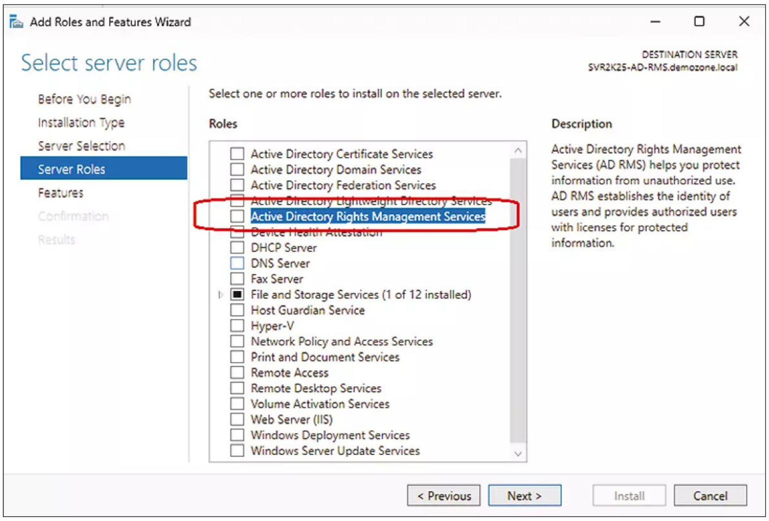
When the AD RMS role is installed and configured, a couple of things happen:

- AD RMS is linked to the domain.
- An RMS cluster URL is created (e.g., *https://rms.demozone.local*).
- A server licensor certificate (SLC) is generated.
- The wizard guides you through the SQL configuration.
- An SSL certificate is installed.
- The RMS service connection point (SCP) is published in Active Directory, which allows domain-joined clients to discover the AD RMS server automatically and apply rules according to rights policy templates created by the AD RMS administrator.

Microsoft Office natively supports AD RMS, so when a Windows client (e.g., a Windows 11 PC with Microsoft Office installed) queries the Active Directory, it will discover the SCP that has been published in Active Directory, thus allowing rules and policies to be applied, which enforces access control and protection of Office files (Microsoft Word, Microsoft Excel, etc.) by rights policy templates.

One of the first tasks that needs to be completed when configuring AD

Lead Image © vska, 123RF.com


Add Roles and Features Wizard
Before You Begin
Installation Type
Server Selection
Server Roles
Features
Confirmation
Results
< Previous
Next >
Install
Cancel

Select server roles

DESTINATION SERVER
SVR2K25-AD-RMS.demozone.local

Select one or more roles to install on the selected server.

Roles

- ☐ Active Directory Certificate Services
- ☐ Active Directory Domain Services
- ☐ Active Directory Federation Services
- ☐ Active Directory Lightweight Directory Services
- ☐ Active Directory Rights Management Services
- ☐ Device Health Attestation
- ☐ DHCP Server
- ☐ DNS Server
- ☐ Fax Server
- ■ File and Storage Services (1 of 12 installed)
- ☐ Host Guardian Service
- ☐ Hyper-V
- ☐ Network Policy and Access Services
- ☐ Print and Document Services
- ☐ Remote Access
- ☐ Remote Desktop Services
- ☐ Volume Activation Services
- ☐ Web Server (IIS)
- ☐ Windows Deployment Services
- ☐ Windows Server Update Services

Description

Active Directory Rights Management Services (AD RMS) helps you protect information from unauthorized use. AD RMS establishes the identity of users and provides authorized users with licenses for protected information.

Figure 1: **Installing the AD RMS role.**

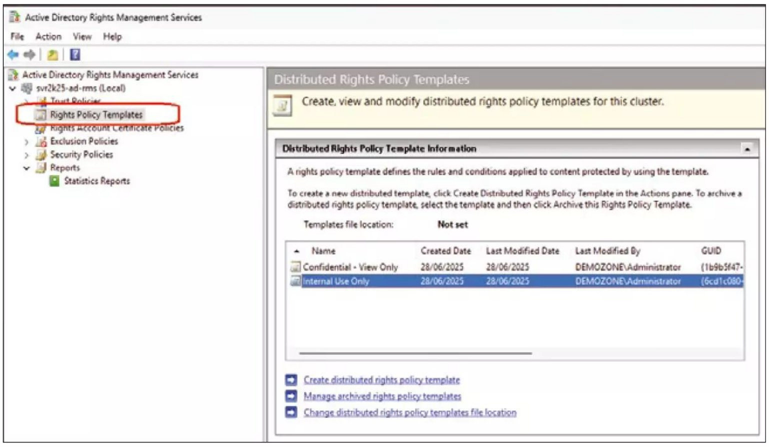
RMS is the initial setup of the relevant rights policy templates for your organization. A distributed rights policy template defines the rules and conditions to be applied to the data that needs to be protected. This step is completed under *Active Directory Rights Management Services* | *Rights Policy Templates* in the management console (**Figure 2**), which is available within the Server Manager utility after the initial installation of the AD RMS role.

Defining multiple different rights policy templates allows a different ruleset to be applied to different classifications of documents or files; for example, a file containing personal data such as a CV or medical record could be defined as highly confidential. Within the rights policy template, then, you could put restrictions in place to lock down the file, permitting only the human resources department to access the file, block the file from being sent by email, or block the printing of the file.

To gain a better understanding of how the process works from an end-user perspective, I'll run through an example of the steps required to apply a rights policy template to a new Word document. In this example, the domain is *demozone.local* and within the domain is a domain controller running Windows Server 2025, a dedicated Windows Server 2025 AD RMS server, and a Windows 11 client with Microsoft Office installed (**Figure 3**).

The following process protects a document with AD RMS:

1. Launch Microsoft Word on the Windows 11 client and create a new Word document.
2. Click *File* | *Info* | *Protect Document* | *Restrict Access* | *Connect to Rights Management Services* (**Figure 4**).
3. The locally installed Microsoft Office client discovers the SCP that was published in Active Directory, connects to the AD RMS server, retrieves the rights policy templates, and presents them to the user to select the appropriate template to apply to the Word document.
4. When the selected template is applied to the document, Microsoft Office encrypts the document with a content key encrypted by the public key of the AD RMS server and with a publishing license that contains rules about (1) who has


Active Directory Rights Management Services
File
Action
View
Help
Active Directory Rights Management Services
svr2k25-ad-rms (Local)
Trust Policies
Rights Policy Templates
Rights Account Certificate Policies
Exclusion Policies
Security Policies
Reports
Statistics Reports
Create distributed rights policy template
Manage archived rights policy templates
Change distributed rights policy templates file location

Distributed Rights Policy Templates

Create, view and modify distributed rights policy templates for this cluster.

Distributed Rights Policy Template Information

A rights policy template defines the rules and conditions applied to content protected by using the template.

To create a new distributed template, click Create Distributed Rights Policy Template in the Actions pane. To archive a distributed rights policy template, select the template and then click Archive this Rights Policy Template.

Templates file location: Not set

Name	Created Date	Last Modified Date	Last Modified By	GUID
Confidential - View Only	28/06/2025	28/06/2025	DEMOZONE\Administrator	{1b9b5f47-
Internal Use Only	28/06/2025	28/06/2025	DEMOZONE\Administrator	{6cd1c080-

Figure 2: **Configuring the AD RMS rights policy templates.**

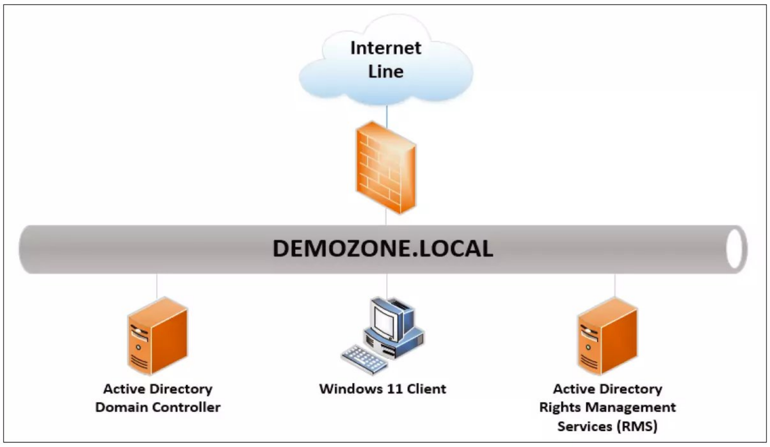
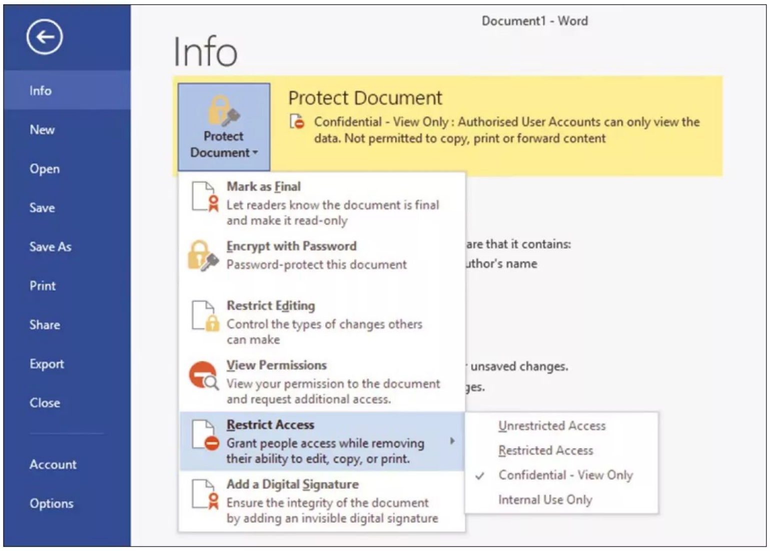

Internet Line
DEMOZONE.LOCAL
Active Directory Domain Controller
Windows 11 Client
Active Directory Rights Management Services (RMS)

Figure 3: **Sample network topology of the structure to support AD RMS.**



Protect Document

Confidential - View Only : Authorised User Accounts can only view the data. Not permitted to copy, print or forward content

Mark as Final Let readers know the document is final and make it read-only

Encrypt with Password Password-protect this document

Restrict Editing Control the types of changes others can make

View Permissions View your permission to the document and request additional access.

Restrict Access Grant people access while removing their ability to edit, copy, or print.

Add a Digital Signature Ensure the integrity of the document by adding an invisible digital signature

Figure 4: Connecting to AD RMS to retrieve the rights policy to be applied to the document.

access to the file, (2) the rights allowed on that file (e.g., view, edit, print), (3) the expiration policies, and (4) an encrypted content key.

5. The document is saved in a protected format with the rules from the rights policy templates applied.

The following process opens a document protected by AD RMS:

1. If another user attempts to open the protected Word document, the Office client sends a request to the AD RMS server, authenticates the user against Active Directory, and checks the publishing license applied at the time the document was protected to determine whether or not they have the appropriate permissions.
2. Assuming the user has the appropriate permissions, the AD RMS server issues a use license to the user, thus granting them permission to access the file.
3. The Microsoft Office application opens the file according to the permissions set out in the rights policy template that was originally assigned to the document.

The key takeaway with regard to AD RMS, particularly in relation to DLP, is that the entire process is reliant on real-time access to the AD RMS server. If the AD RMS server is not available, the publishing license cannot be verified; hence, the file cannot be opened. Therefore, if someone were to email that file to a recipient outside of your organization, that person will be outside of your network and will not be able to connect to the rights management server to check whether that person has rights to open the file, resulting in that user not being permitted to open the file.

AD RMS can be integrated with existing in-house, on-premises file servers, and policy templates can be applied automatically by installing the File Server Resource Manager role on the relevant file server and configuring the File Server Resource Manager to connect to the AD RMS server, pointing it to the relevant folders to which you want to apply the rights policy templates. You can then schedule a policy for when to discover files within the configured folders and apply the rights policy templates to the files on a schedule.

Table 1: Events

Event ID	Description
100	A licensing request was processed.
101	A publishing license was created.
102	Rights account certificate was issued.
1200	A user authenticated successfully with AD RMS.
1202	A user failed to authenticate with AD RMS, which could indicate a brute-force attack as multiple events occurred in a short period of time.
4670	Permissions on an object were changed.

Logging

AD RMS writes important information to the application, security, system, and AD RMS event logs on the AD RMS server. Some specific events that are critical to monitor are shown in Table 1.

Although you can manually monitor these event logs, you could also export them to a security information event management (SIEM) platform and use that platform to analyze, correlate, and report on log and event data to detect any abnormal access to files protected by AD RMS. You could then configure the SIEM to send you a daily or weekly operational report, or indeed you could configure the SIEM to alert you on any critical events that occur so that you can proactively monitor for potential data loss incidents.

Conclusion

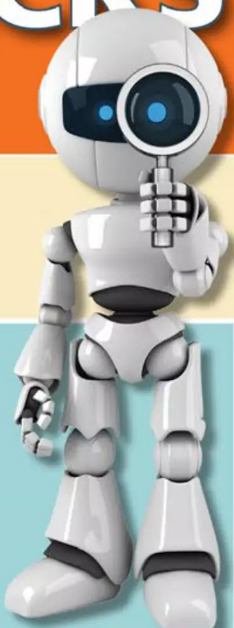
Protecting data within an organization is a core responsibility for any data owner. Although cloud-orientated organizations will look toward solutions such as Azure Rights Management to protect data in the cloud, if your data currently resides within on-premises file servers, AD RMS could offer an opportunity to secure your Microsoft Office data and become part of a wider DLP strategy for your organization. ■

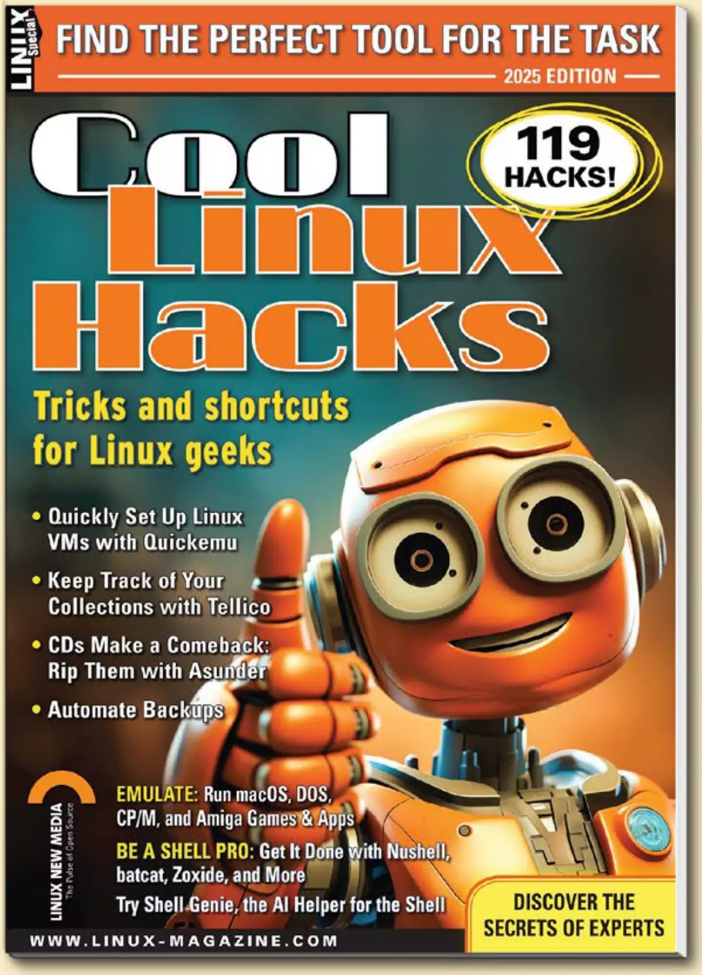
Author

Conor Fitzgerald is an IT professional with over 20 years of knowledge and expertise. He is keenly interested in all aspects of IT and is a huge advocate of finding real-world solutions to modern day technological challenges.

SHOP THE SHOP
shop.linuxnewmedia.com/shop

GET PRODUCTIVE WITH
COOL LINUX HACKS




LINUX Special
FIND THE PERFECT TOOL FOR THE TASK
2025 EDITION
Cool Linux Hacks
119 HACKS!
Tricks and shortcuts for Linux geeks
• Quickly Set Up Linux VMs with Quickemu
• Keep Track of Your Collections with Tellico
• CDs Make a Comeback: Rip Them with Asunder
• Automate Backups
EMULATE: Run macOS, DOS, CP/M, and Amiga Games & Apps
BE A SHELL PRO: Get It Done with Nushell, batcat, Zoxide, and More
Try Shell Genie, the AI Helper for the Shell
DISCOVER THE SECRETS OF EXPERTS
LINUX NEW MEDIA
The Pulse of Open Source
WWW.LINUX-MAGAZINE.COM

Improve your Linux skills with this cool collection of inspirational tricks and shortcuts for Linux geeks.

- Google on the Command Line
- OpenSnitch Application Firewall
- Parse the systemd journal
- Control Git with lazygit
- Run Old DOS Games with DOSBox
- And more!

ORDER ONLINE:
shop.linuxnewmedia.com/shop

Session cookies as targets for criminals

Stealing from the Cookie Jar

Session data stored on user hard disks can be targeted by criminals to bypass the need for a password or second factor in active user sessions. By Matthias Wübbeling

In the early days of the Internet, JavaScript and cookies were considered evil and dangerous constructs that could be misused to execute code and secretly store tracking information. Users who valued security disabled JavaScript and simply banned cookies, with few exceptions. Today, both concepts are an integral part of the Internet. Responsive web applications can be created quickly with the use of comprehensive libraries and APIs on the server for all communication. Static API keys, JSON web tokens (JWTs), or simply session cookies, which are essentially dynamic API keys, are used to secure user activities.

Session cookies are small snippets of information that web applications use to reference temporary information about a user's session, making it possible to identify a user visiting a website without the need constantly to re-authenticate. Typical session information includes login status, shopping cart content in online stores, and user configurations. As such, session cookies form the basis for web applications.

Session cookies are normally stored in the web browser's cache on the user device, and you can view them with developer tools (Figure 1). Unlike permanent cookies, which the browser also stores on disk, session cookies only need to be stored in the browser's working memory. You can access this memory area from the browser's `sessionStorage` API. The information is then only available while the browser is running. As soon as the user closes the browser window, the data disappears automatically, which makes the approach more secure because no data remains permanently on the device. However, many applications also store their data cookies in the browser's local cookie store so that they are also available in other windows of the same application.

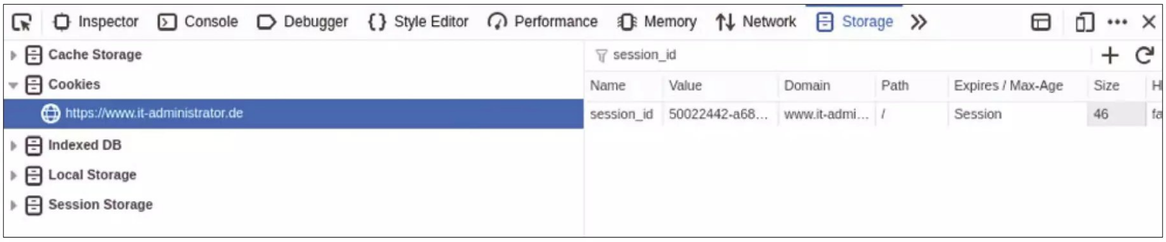
This kind of storage is not a problem as long as the validity is limited to the duration of the browser session. With JWTs, validity times can be implemented directly in the token to guarantee automatic expiration.

Even though session cookies are fairly short-lived compared with their permanent counterparts (after all, they do not need to store data across sessions and are limited to the duration of a single website visit), their validity does not depend solely on the website in the browser.

Expiry Dates at a Glance

The lifetime of a session on the server side is determined by the server configuration of the web application, regardless of the actual session cookies in the browser. A session can therefore continue even after the matching cookie has been deleted in the browser, as long as the server does not flag the associated session record as expired. This lifetime is usually defined by a timeout value set by the server logic. Typically, the runtime is based on the duration of a typical user session, but tends to be longer or is automatically extended with each use so as not to disturb users.

The Open Worldwide Application Security Project (OWASP) Foundation recommends a lifetime of two to five minutes for applications with high security requirements, such as online banking. For all other applications,



Name	Value	Domain	Path	Expires / Max-Age	Size	H
session_id	50022442-a68…	www.it-admi…	/	Session	46	fa

Figure 1: Displaying the local storage of a web application in developer mode.

Photo by Vyshnavi Bisani on Unsplash

the suggested timeout value is 15 to 30 minutes for the cookie to expire after inactivity [1]. However, even for active use, OWASP recommends a defined time for the session to expire according to typical everyday use (i.e., approximately every four to eight hours in a professional context). You will find out why this is important in a moment.

After this defined period has expired, the server then deletes the session or invalidates it – regardless of the cookie in the web browser – and the user would then have to log in again. For reasons of user-friendliness, large providers in particular sometimes use cookies that last far longer. On Google, Facebook, and Amazon, for example, these cookies are sometimes valid for several months and are initialized as device cookies with different values on different devices. In the device or session overviews, you as a user can then log out of devices and explicitly end sessions.

Cookies in the Wrong Hands

In recent years, cases of leaked user login credentials have skyrocketed. As a result, increasing numbers of web application providers are introducing additional factors for user authentication. The best-known mechanism is probably the use of authenticator applications such as Authy, Google, Microsoft, FreeOTP, and so on. These authenticators require users to enter a computed token in addition to their email address and password during the authentication process. The (sometimes incorrect) use of passkeys as a replacement for classic login data can also significantly restrict the success of these attacks, which go under the generic name of account hijacking.

Session cookies then become all the more interesting to criminals. Once in possession of a valid cookie, the user's authenticated session can simply be continued on the attacker's computer. Neither login data, MFA tokens, or passkeys are then required to hijack the account. Criminals can obtain session cookies in various ways. Hackers' methods are often based on security vulnerabilities in web applications and user behavior.

One technique often discussed in awareness training courses is the man-in-the-middle attack, in which attackers intercept data traffic on unsecured networks, such as public WiFi networks in hotels or cafés. Because almost all web applications now use TLS-secured connections, this approach is rarely successful today.

Another method is known as cross-site scripting (XSS). Malicious code is injected into a website and, when executed in the victim's browser, can read and forward session cookies.

Social engineering and phishing are often used to trick users into revealing sensitive information, such as the session ID contained in a URL.

Session fixation is a particularly insidious attack. Attackers create their own sessions in the application but do not log in. They then distribute a manipulated link that contains the session ID of the session in the URL. If a victim follows this link and then logs in to the web application later on, the attacker's session is converted into an authenticated session of the victim with the use of the session ID at that moment.

Malware as the Cause

Recently, stolen data records from malware infections have repeatedly been leaked to the public. The data comes directly from users' computers and sometimes contain login details and session cookies. Users accidentally infect their computers with malware through insecure downloads or websites, and the malware then continuously logs the user's actions and inputs. The password manager content is, of course, a primary target for attackers. With the master password, which is of course read when entered, an attacker can quickly obtain many of a victim's login credentials with little effort. However, if authentication is secured with additional factors as described above, the attacker cannot do much with the credentials at first.

Customized malware in the form of a cookie stealer is designed both to read the typical user data and grab cookies directly from the browser. While the victim is actively using the web application, the window of opportunity for criminal activity remains large enough. Even the FBI has published an official warning about account hijacking with the use of session cookies [2]. Browser and web application manufacturers are working on appropriate countermeasures. Google, in particular, is attempting to further restrict use on other computers by means of unique device-specific cookies.

Conclusions

Session cookies store session information from websites and often the user's authentication details, which makes these cookies interesting for criminals, who can use them to bypass multifactor authentication. As always, special caution is advised. ■

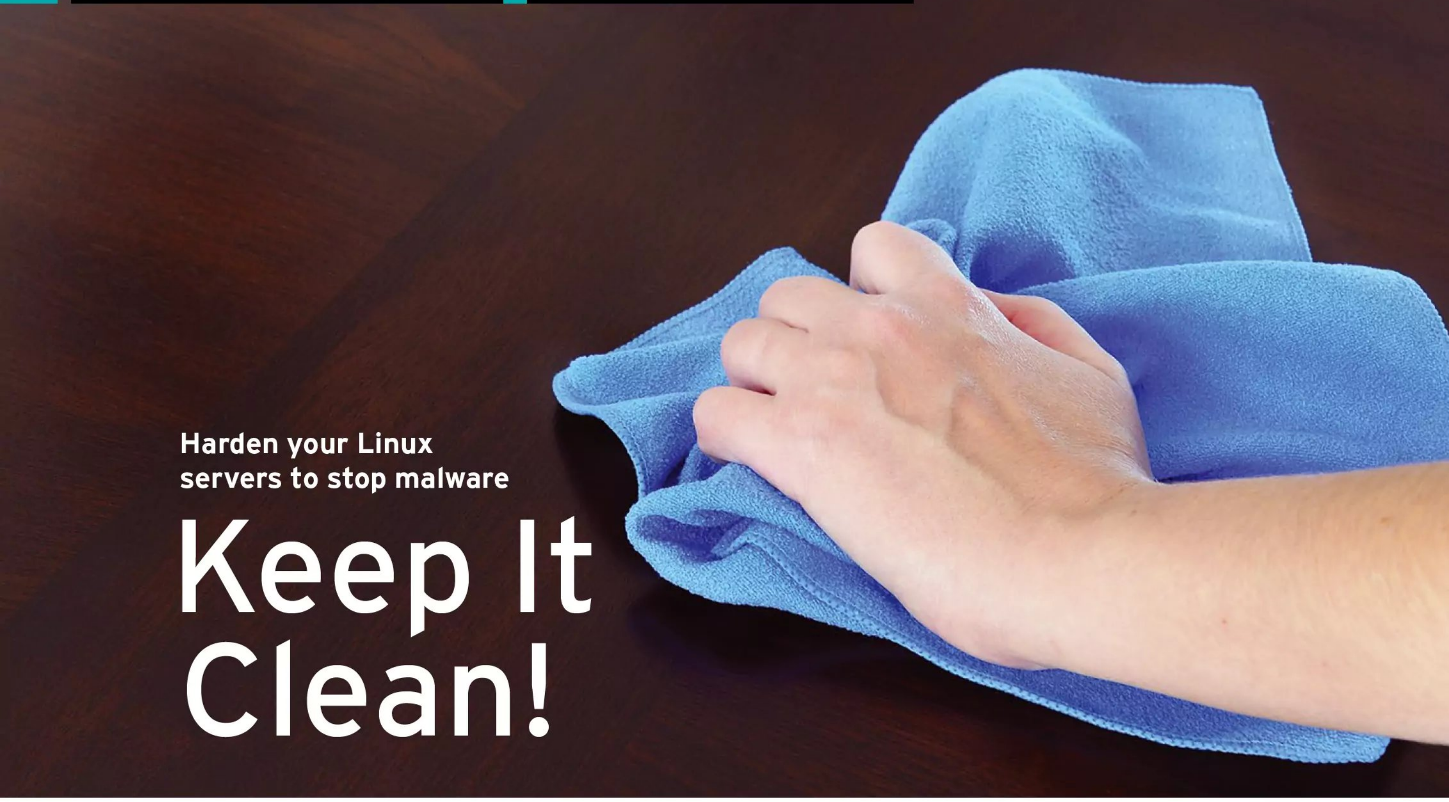
Info

[1] OWASP recommendations on session management: [https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html#Session_Expiration]

[2] FBI warning on account takeover: [https://www.fbi.gov/contact-us/field-offices/atlanta/news/cybercriminals-are-stealing-cookies-to-bypass-multifactor-authentication]

The Author

Dr. Matthias Wübbeling is an IT security enthusiast, scientist, author, consultant, and speaker. As a Lecturer at the University of Bonn in Germany and Researcher at Fraunhofer FKIE, he works on projects in network security, IT security awareness, and protection against account takeover and identity theft. He is the CEO of the university spin-off Identeco, which keeps a leaked identity database to protect employee and customer accounts against identity fraud. As a practitioner, he supports the German Informatics Society (GI), administrating computer systems and service back ends. He has published more than 100 articles on IT security and administration.



Harden your Linux servers to stop malware

Keep It Clean!

Delve into the software treasure chest to find useful products that improve virus protection on Linux. By Markus Stubbig

Malware programmers can target significantly more potential victims by developing and distributing their products for Windows, so consumer protection organizations often do not mention antivirus products for Linux. This oversight does not mean the free operating system is secure. Although viruses, worms, and Trojans are less common on Linux than on Windows, they still pose a threat.

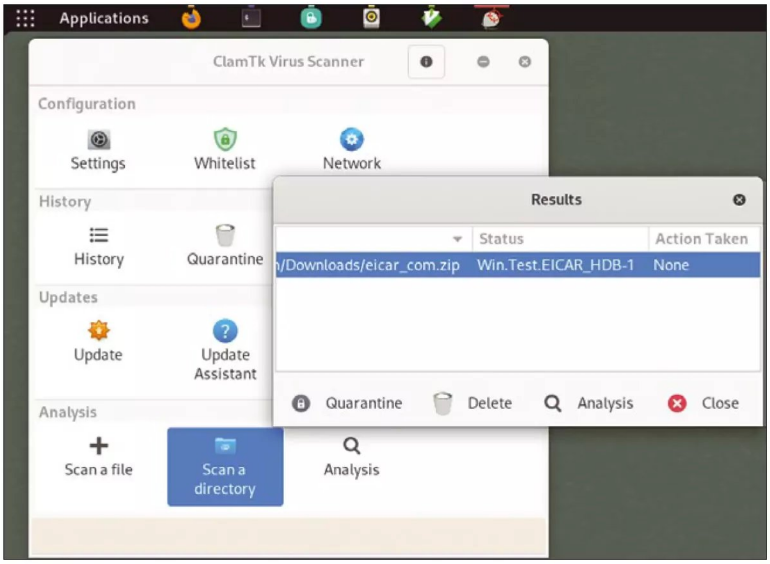
For this reason, I look at some projects here that provide basic protection against malware for your Linux server with very little overhead. If your server is a file or mail server, the protection is even indirectly extended to connected (Windows) clients. If you already have basic protection on your Linux boxes, you can also check them for vulnerabilities with a security scanner. For environments in which high security is the order of the day, a security information and event management (SIEM) system is a must-have, which I also look at briefly.

ClamAV

The ClamAV [1] antivirus scanner is probably familiar to every Linux administrator. The software has been around since 2001 and protects Linux and Windows computers alike. In 2013, this classic tool was acquired by commercial network equipment supplier Cisco, which has not changed the open source nature of the project. Installation and operation are easy to explain: Use the package manager to install the software, load the signatures, and scan the hard drive. These steps are accomplished on Rocky Linux in just three lines:

```
dnf install clamd clamav
freshclam
clamscan /home
```

Numerous online tutorials can help you set up ClamAV. As a bonus, ClamTk (Figure 1) and ClamWin even offer graphical front ends. Next, you can turn your attention to two little-used features: unofficial signatures and real-time protection.


Applications
ClamTk Virus Scanner
Configuration
Settings
Whitelist
Network
History
History
Quarantine
Updates
Update
Update Assistant
Analysis
Scan a file
Scan a directory
Analysis
Results
Status
Action Taken
Quarantine
Delete
Analysis
Close

```
/Downloads/eicar_com.zip   Win.Test.EICAR_HDB-1   None
```

Figure 1: The ClamTk virus scanner graphical interface is minimalistic but includes the most important functions.

Lead Image © Danila Krylov, 123RF.com

Obtaining the Latest Signatures

The `freshclam` command in the previous code accesses predefined signature servers, retrieves the latest changes and stores them in the local database. However, this process is not the only way to source virus patterns. Providers such as URLhaus and Sanesecurity offer additional signatures. If you don't want to check the various providers manually, you can use the somewhat outdated ClamAV Unofficial Signature Updater [2] package or the Fangfrisch [3] tool, which specialize in this task. Fangfrisch retrieves the community signatures and feeds them to `freshclam`, where they ultimately reach ClamAV. Python's package installer, `pip`, installs the service on your Linux computer. The configuration file in Listing 1 tells Fangfrisch in which provider ocean to fish. Freshclam is then notified of the new pattern source:

```
cat <<EOF >> /etc/freshclam.conf
DatabaseCustomURL ➜
  file:///var/lib/fangfrisch/db.sqlite
EOF
freshclam
```

Not all providers of antivirus (AV) signatures are free. SecuriteInfo, for example, charges an annual fee for its services. The free basic account releases the signatures after a delay.

Real-Time Virus Scanning

With a database primed with virus patterns, ClamAV can even check your data in real time. This feature is known as on-access scanning, and the name says it all. ClamAV has to give its approval before a file can be accessed locally. If the file fails the check, it cannot be opened.

You can implement real-time protection in two ways: with the `clamonacc` scanner and the ClamFS filesystem. Both tools work closely with the `clamd` service, which stores the virus definitions and checks everything sent by `clamonacc` and ClamFS.

The `clamonacc` scanner liaises with the kernel and waits for information about desired file access. The two services communicate over a Unix socket or a TCP/IP connection. Two points are crucial for configuration: Be sure to exclude the ClamAV user from the scan to avoid endless loops, and only scan selected areas – not everything below the root (`/`) directory. Both services use the same configuration file, `/etc/clamd.d/scan.conf`. The following settings tell ClamAV to monitor the users' home directories and prevent access if a virus is suspected:

```
LocalSocket /run/clamd.scan/clamd.sock
OnAccessExcludeUname clamscan/
OnAccessIncludePath /home
OnAccessPrevention yes
```

Now restart both services; `clamd` will need a few seconds to load all the AV signatures into memory:

```
systemctl restart clamd@scan
systemctl restart clamonacc
```

ClamFS is a userspace filesystem and works slightly differently from `clamonacc`. All access to the filesystem mounted by ClamFS must first pass through the virus scanner, which must give its approval.

The configuration (in XML format) requires the socket to `clamd`, a source directory (root), and a directory in which the source directory with AV protection is to be mounted in the filesystem (mountpoint). With a minimal configuration file (`demo.xml`),

```
<clamfs>
  <clamd socket=➜
   "/run/clamd.scan/clamd.sock" />
  <filesystem root="/store/upload" />
  <filesystem mountpoint=➜
   "/var/www/upload/" />
</clamfs>
```

ClamAV protects access to files in the mountpoint directory, which is verified by entering

```
clamfs demo.xml
```

However, this virus protection does not apply to the root folder. Of course, a final test is essential. Working as a regular Linux user, download the EICAR test file:

```
wget --output-document=~/eicar.txt ➜
  https://secure.eicar.org/eicar.com.txt
```

The download should work without any problems and save the file in your home directory. Leave it in your home directory for `clamonacc`, or drop it into `/var/www/upload` for ClamFS. ClamAV or ClamFS only becomes active when you try to access the file, preventing the read operation with a simple *Operation not permitted* message. At the same time, an entry is logged in the journal indicating that the EICAR signature has been detected. This behavior differs from that of other virus scanners that immediately delete or quarantine everything at the slightest suspicion.

Because ClamAV is available in most repositories, basic protection can be set up and enabled with just a few commands to your distribution's package manager. However, decades of malware development have also lead to other malicious programs – such as rootkits.

Fear Not Rootkits

Rootkits are small pieces of software that hide on a system and conceal processes or files. For example, a rootkit can modify the ps command so that it hides certain entries from the process list. In its simplest form, this rootkit can be constructed with an alias:

```
alias ps='_ps() { ➜
  ps $1 | grep --invert-match virus; ➜
  }; _ps'
```

The `alias` command filters the output and removes any line containing the

Listing 1: Fangfrisch Config File

```
pip install fangfrisch
wget --output-document=fangfrisch.conf https://raw.
        githubusercontent.com/rseichter/fangfrisch/
        master/docs/example.conf
mkdir --mode=0770 --parents /var/lib/fangfrisch
chgrp virusgroup /var/lib/fangfrisch
fangfrisch --conf fangfrisch.conf initdb
fangfrisch --conf fangfrisch.conf refresh
```

word *virus*. In the background, the virus process continues to run unnoticed and wreaks havoc. The `ps` rootkit is not very effective, because it can be easily circumvented with commands such as `top` or by looking in the `/proc` filesystem.

However, clever rootkits nest themselves into the kernel and conceal the existence of the virus process there, which is where Rootkit Hunter [4] comes in handy, searching the Linux computer for some 80 different rootkits and anomalies in the existing system commands. To install `rkhunter` on your system, enter the command

```
dnf install rkhunter
rkhunter --update
rkhunter --check --skip-keypress
```

In this example, `rkhunter` finds the Opyum kit component. A glance at the logfile immediately reveals that it is the `/bin/i` file. Unfortunately, the mere existence of this file is enough to trigger an alarm for `rkhunter`, even though it is only a soft link to `/bin/grep` in this case.

Rkhunter does not have real-time protection, but it does have a cron function: When

```
rkhunter --cronjob
```

is called, the tool does not prompt for input or display colors, and it sends the output to the logfile. The last 20 lines contain a summary, and the *Possible rootkits* line tells you whether the system is infected.

Middle Ground: Linux Malware Detect

Linux Malware Detect (LMD) [5] is a kind of middle ground between rootkits and antivirus scanners. The project collects information from various intrusion detection systems and uses it to build a signature database. The software is not available as a ready-to-run package in any repository, so you will need to use the tarball from GitHub to install, and because LMD comprises only Bash and Perl scripts, you do not need a compiler:

```
dnf install ed perl inotify-tools
git clone https://github.com/rfxn/↩
  linux-malware-detect.git
cd linux-malware-detect/
./install.sh
systemctl enable --now maldet
```

The `systemctl` command starts monitoring mode, which keeps an eye on logged-in users. You can check the filesystem by typing

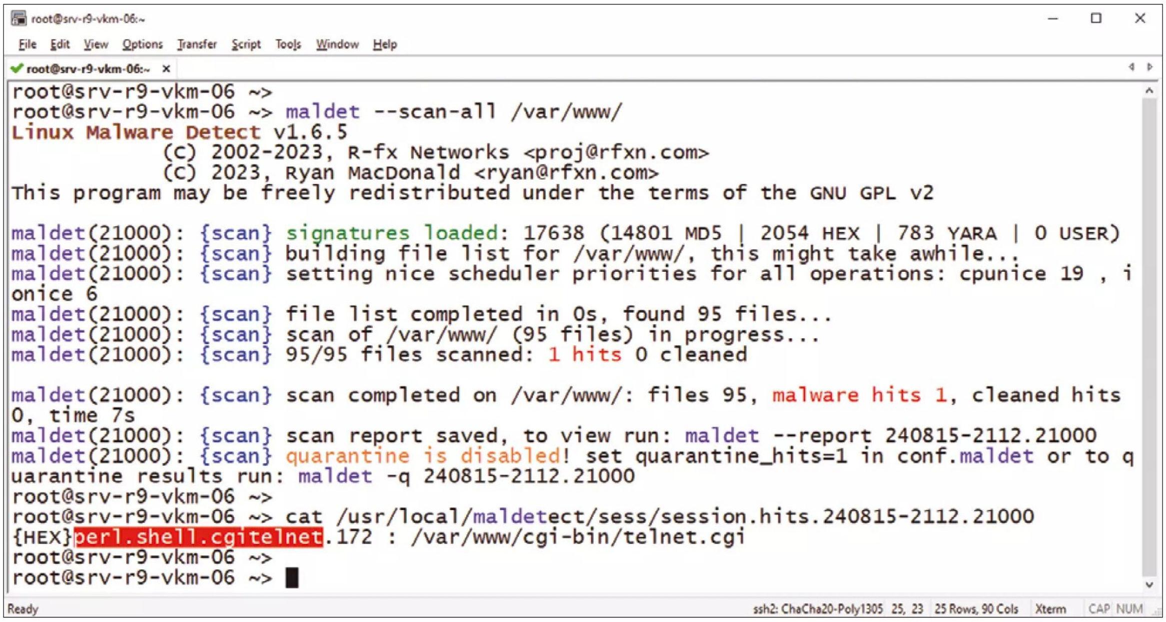
```
maldet --scan-all /
```

After launching, the scan process assigns itself the lowest CPU and I/O priority in order to not slow down the system. The process is surprisingly fast, because LMD only checks files that do not belong to the root user. If you want to check all files, you can set `scan_ignore_root="0"` in the `conf.maldet` configuration file to check the entire database.

When `maldet` needs to scan large areas of the filesystem, it is noticeably slow. To speed up the process, LMD can cooperate with ClamAV by transferring its signatures to FreshClam and telling the `clamscan` process or the `clamd` server to check the files. The settings you need are well documented in the configuration file, but the AV scan is only sped up if the `clamd` server has sufficient CPU power and RAM. Figure 2 shows that LMD detected malware in the web server directory. By default, it just displays an alert with instructions on how to quarantine the file.

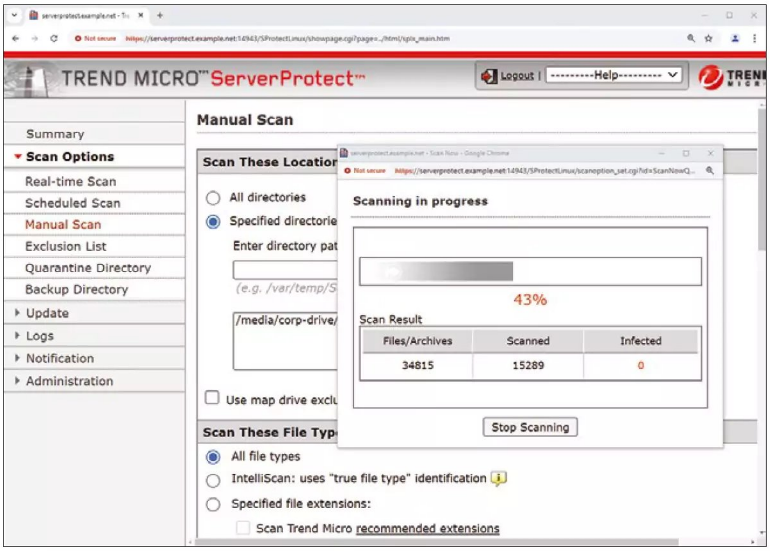
Commercial Solution: ServerProtect

In addition to all of the open source software, Trend Micro offers the


root@srv-r9-vkm-06:~
File Edit View Options Transfer Script Tools Window Help
root@srv-r9-vkm-06:~
Ready
ssh2: ChaCha20-Poly1305 25, 23 25 Rows, 90 Cols Xterm CAP NUM

```
root@srv-r9-vkm-06 ~>
root@srv-r9-vkm-06 ~> maldet --scan-all /var/www/
Linux Malware Detect v1.6.5
            (C) 2002-2023, R-fx Networks <proj@rfxn.com>
            (C) 2023, Ryan MacDonald <ryan@rfxn.com>
This program may be freely redistributed under the terms of the GNU GPL v2

maldet(21000): {scan} signatures loaded: 17638 (14801 MD5 | 2054 HEX | 783 YARA | 0 USER)
maldet(21000): {scan} building file list for /var/www/, this might take awhile...
maldet(21000): {scan} setting nice scheduler priorities for all operations: cpunice 19 , i
onice 6
maldet(21000): {scan} file list completed in 0s, found 95 files...
maldet(21000): {scan} scan of /var/www/ (95 files) in progress...
maldet(21000): {scan} 95/95 files scanned: 1 hits 0 cleaned

maldet(21000): {scan} scan completed on /var/www/: files 95, malware hits 1, cleaned hits
0, time 7s
maldet(21000): {scan} scan report saved, to view run: maldet --report 240815-2112.21000
maldet(21000): {scan} quarantine is disabled! set quarantine_hits=1 in conf.maldet or to q
uarantine results run: maldet -q 240815-2112.21000
root@srv-r9-vkm-06 ~>
root@srv-r9-vkm-06 ~> cat /usr/local/maldetect/sess/session.hits.240815-2112.21000
{HEX}perl.shell.cgitelnet.172 : /var/www/cgi-bin/telnet.cgi
root@srv-r9-vkm-06 ~>
root@srv-r9-vkm-06 ~> █
```

Figure 2: When Linux Malware Detect detects malware, it immediately prompts you to quarantine the file.



TREND MICRO™ ServerProtect™

Logout | ---------Help---------

Manual Scan

Summary

Scan Options

Real-time Scan

Scheduled Scan

Manual Scan

Exclusion List

Quarantine Directory

Backup Directory

Update

Logs

Notification

Administration

Scan These Locatior

○ All directories

● Specified directorie

Enter directory pat

(e.g. /var/temp/S

/media/corp-drive/

☐ Use map drive exclu

Scan These File Typ

● All file types

○ IntelliScan: uses "true file type" identification

○ Specified file extensions:

☐ Scan Trend Micro recommended extensions

Scanning in progress

43%

Scan Result

Files/Archives	Scanned	Infected
34815	15289	0

Stop Scanning

Figure 3: Trend Micro ServerProtect is aimed at commercial customers and can be conveniently configured in a web GUI.

ServerProtect for Linux product, which is an AV scanner with real-time protection. Users wanting to try out the tool can use a free one-month license. A web server is automatically included for configuration, and it can be used without too much of a learning curve. All of the tool's actions can be triggered from the web user interface and the command line.

A scheduled scan periodically checks selected folders, and a manual scan kicks in when needed. For real-time scanning, Trend Micro provides a kernel module that is precompiled for Red Hat and SUSE distributions. Because the prebuilt modules tend to use older kernels, you will need to compile for more recent Linux versions. In principle, it will work with any kernel, but in practice, you might get stuck. Either the distribution is not recognized or dubious compiler errors block the path to the kernel module, and with Secure Boot enabled, the kernel also expects a cryptographic signature from the module before the real-time scan can begin.

If you manage to create a working kernel module, your reward is a real-time scanner. For everyone else, you still have the scheduled and manual scans (Figure 3). Signature updates are straightforward and run smoothly in the background. However, ServerProtect's documentation reveals an outdated impression. The installation steps use the `chkconfig` and `initd` commands, which many Linux distributions replaced with systemd more than a decade ago, and information about kernel module signatures is nowhere to be found.

The target audience is clearly not individual desktop computers, but servers. If you want to organize your server farm's virus protection centrally, you can manage the ServerProtect instances with a Control Manager agent or an Apex Central agent. Ironically, these software products are only available for Windows.

Trend Micro charges a hefty price for its protection but offers solid virus defense with a convenient website for configuration. The manufacturer is cautious about costs and leaves pricing to its partners.

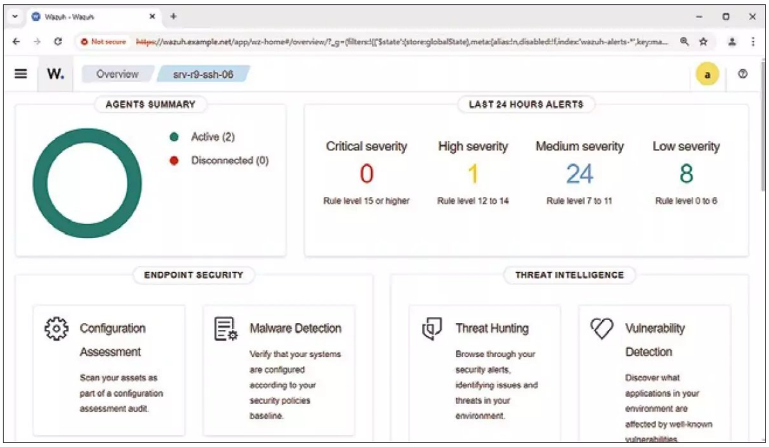
Wazuh Open Source SIEM

Wazuh is a fully-fledged SIEM platform that puts an end to the "just install it" principle. The basic idea is that an agent runs on each end device and reports to the Wazuh server. The server analyzes the information, creates an overview, and assesses the security status of the infrastructure.

For a first impression or in a small environment, Wazuh provides a virtual machine [6] that contains all services and appears at *https://<Server IP Address>* after booting. Without agents, though, the web pages will be largely empty. The *Deploy new agent* button shows how the agent software is installed on clients for various operating systems and distributions.

Nothing happens immediately after rolling out the agents. Wazuh requires patience, because the agents perform their malware and integrity checks on a schedule. Gradually, the dashboards on the Wazuh website fill up, providing information on the health status (Figure 4).

Wazuh describes itself as an open source security platform [7] and goes far beyond basic protection, with a focus on a holistic view of all endpoints



Overview srv-r9-ssh-06

AGENTS SUMMARY

Active (2)

Disconnected (0)

LAST 24 HOURS ALERTS

Critical severity	High severity	Medium severity	Low severity
0	1	24	8
Rule level 15 or higher	Rule level 12 to 14	Rule level 7 to 11	Rule level 0 to 6

ENDPOINT SECURITY

Configuration Assessment

Scan your assets as part of a configuration assessment audit.

Malware Detection

Verify that your systems are configured according to your security policies baseline.

THREAT INTELLIGENCE

Threat Hunting

Browse through your security alerts, identifying issues and threats in your environment.

Vulnerability Detection

Discover what applications in your environment are affected by well-known vulnerabilities.

Figure 4: Wazuh collects reports from all agents to paint an overall picture of the security situation. Here, the agent has detected and reported several vulnerabilities.

rather than on malware protection for a single computer.

Conclusions

Even though nasty viruses mainly attack Microsoft systems, Linux is not automatically off the hook. The smaller number of malware programs (Table 1) is reflected in the restricted selection of AV products (see the "Even More Security" box). The first port of call is ClamAV, which is available for virtually every distribution and has been sniffing around local databases for more than 20 years. If you want to protect your environment against rootkits, you will want Rootkit Hunter or Linux Malware Detect to scan your systems on a regular basis. Going one step further, to ensure a uniform level of protection on all computers, you can look into security platforms such as Wazuh. ■

Even More Security

The list of exciting projects is far from over, but their areas of application overlap in some cases, such as tools for checking file integrity (e.g., Tripwire, Integrit, and AIDE). These tools create checksums for each system file and store them in a database, repeating the process at regular intervals and comparing the current and previous checksums. If the values do not match, something has changed in the file content, and it is up to the administrator to determine whether it is malware or an update.

Security auditors such as Lynis, Trivy, and OpenVAS take a different approach: They compare the software versions and settings of the existing Linux system with best practices and report any deviations. After completing its scan, Lynis is very helpful and provides specific recommendations (e.g., determining whether protocol `rds` (reliable datagram sockets) is really needed on the system).

Info

[1] ClamAV: [https://www.clamav.net]
[2] ClamAV Unofficial Signatures Updater: [https://github.com/extremeshok/clamav-unofficial-sigs]
[3] Fangfrisch: [https://rseichter.github.io/fangfrisch/]
[4] Rootkit Hunter: [https://rkhunter.sourceforge.net]
[5] Linux Malware Detect: [http://www.rfxn.com/projects/linux-malware-detect/]
[6] Wazuh VM: [https://documentation.wazuh.com/current/deployment-options/virtual-machine/virtual-machine.html]
[7] Wazuh: [https://wazuh.com]

The Author

Markus Stubbig is a networking engineer who has worked in the IT industry for 20 years. His strong focus is on the design and implementation of campus networks around the world.

Table 1: Linux Anti-Malware Tools

	ClamAV	Linux Malware Detect	Rootkit Hunter	ServerProtect
Version	1.4.0	1.6.5	1.4	3.0
Release year	2001	2009	2006	2005
Language	C/C++	Shell/Perl	Shell/Perl	-
Installation complexity	Easy	Low	Easy	High
Complexity of use	Easy	Moderate	Moderate	Easy
Real-time scan	Y	Y	N	Y
Documentation	+	+	+	++
Open source	GPLv2	GPLv2	GPLv2	No
Native/Docker	Yes/Yes	Yes/No	Yes/No	Yes/No
Commercial support	Yes	No	No	Yes

Zero Trust
grommunio
Storage
ADMIN
Network & Security
STORAGE
Developments, management, and new tech
Deduplication on Windows Server 2022
grommunio
Drop-in replacement for Microsoft Exchange
Zero Trust
Planning and implementation
Non-Relational Databases
Monster VMs
Gitea
ADMIN
Network & Security
Non-Relational Databases
Purdue Model
Kubeflow
Monitoring
ADMIN
Network & Security
Monitoring
Networks, applications, and long-term storage
Bloonix
Web-based network monitoring
Kubeflow
Kubernetes setup for AI development
Data Obfuscation
Secure AD Design
Flipper Zero
ADMIN
Network & Security
Data Obfuscation
Anonymize and pseudonymize your data
OpenFaaS
Lightweight Kubernetes
AD Risk Mitigation
IVRE
ADMIN
Network & Security
Lightweight Kubernetes
3 compact Kubernetes compared
Active Directory Risk Mitigation
IVRE
Visualize IT infrastructures
Software Licenses
Coroot
Monitoring and attack detection with eBPF
Analyze and Optimize File I/O
NAPS2 SDK
Develop a simple app for scanning documents
dockle
Keep your Docker containers compliant
Admin Distros
Local AI Server
A New Checkmk
ADMIN
Network & Security
5 Network Admin Distros
ALT Linux Rescue, Finnix, Grml, Knoppix, and SystemRescue
Local AI Server
Retain data sovereignty with an in-house AI server
Snapshots in Azure
Checkmk
A new open source monitoring implementation for network services
PipeCD
A lean CI/CD orchestrator for Kubernetes and other platforms
Secure AI
SELinux secures AI model deployments in real-world scenarios

REAL SOLUTIONS FOR REAL NETWORKS

ADMIN is your source for technical solutions to real-world problems.

Improve your admin skills with practical articles on:

- Security
- Cloud computing
- DevOps
- HPC
- Storage and more!

GET IT FAST
with a digital subscription!

6 issues per year!

ORDER NOW

shop.linuxnewmedia.com/shop

ADMIN
Network & Security

ADMIN magazine

@adminmag

@adminmagazine@hachyderm.io

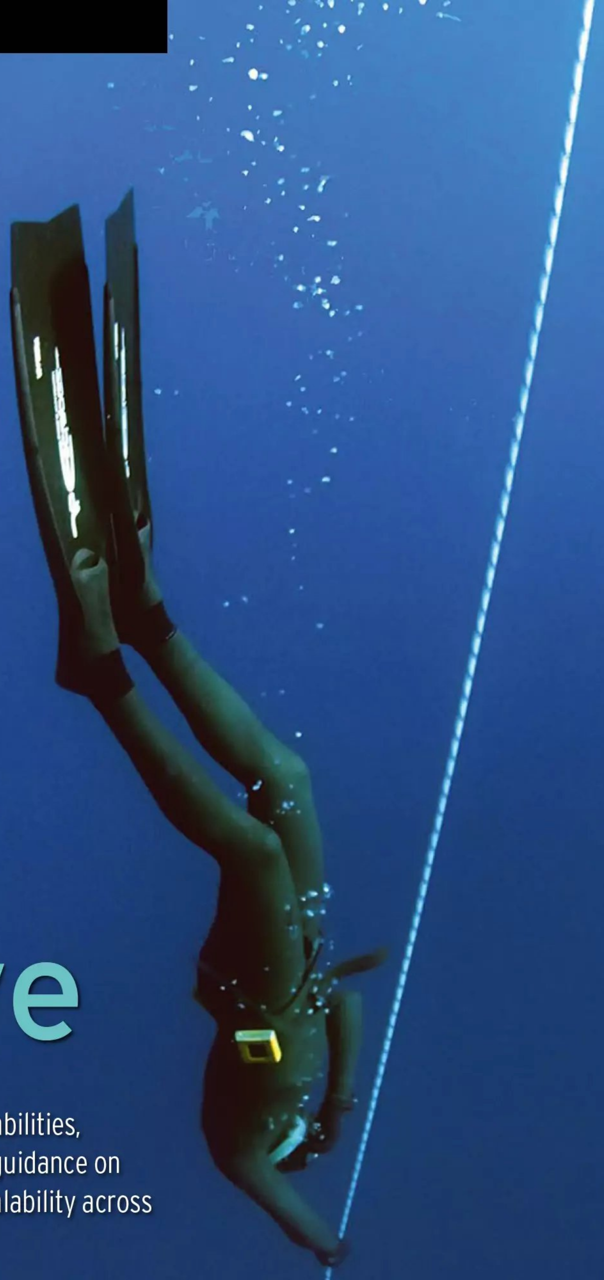


Photo by Jakob Boman on Unsplash

Full-spectrum security scanner

Deep Dive

We take a close look at the Trivy scanners for vulnerabilities, misconfigurations, and secrets with Ubuntu-centric guidance on performance tuning, security configurations, and scalability across Linux distributions. By Marcin Gastol

Trivy is an open source, all-in-one security scanner by Aqua Security that has rapidly become a go-to tool in the DevSecOps toolkit [1]. It earned this reputation by combining multiple security checks such as vulnerability scanning, configuration auditing, and secret detection into a single easy-to-use command-line interface.

With a single binary and minimal setup, Trivy can be run on virtually any Linux distribution and integrated into continuous integration and continuous delivery (CI/CD) pipelines or cloud environments without friction. Its popularity is evident from a vibrant community (tens of thousands of GitHub stars) [2] and its adoption as the default scanner in platforms such as Harbor (the Cloud Native Computing Foundation (CNCF) image registry).

In an era of rising software supply chain threats, Trivy's approach addresses the need for continuous security checks at every stage of development and deployment. I walk you through Trivy's capabilities, how it works under the hood, and how to configure and optimize it on Linux for performance, security, and scale.

Main Capabilities

One reason Trivy is so valued is its breadth of coverage. It can scan container images (both local images and those in registries), the filesystems of running systems or virtual machines (VMs), remote code repositories, infrastructure-as-code (IaC) manifests, and even live Kubernetes clusters.

In practice, this means a single tool can examine everything from your base operating system packages and application libraries to Dockerfiles, Terraform templates, and Kubernetes YAMLs, searching for any signs of security issues.

Trivy's primary function is to detect common vulnerabilities and exposures (CVEs) in software components. When pointed at a container image or a filesystem, it identifies vulnerable operating system packages (e.g., in Debian, Alpine, Ubuntu, RHEL, etc.), as well as vulnerabilities in application dependencies (language-specific packages like Pip/Pipenv for Python, npm/Yarn for Node.js, Maven for Java, etc.).

Under the hood, Trivy aggregates data from multiple security advisories and feeds (including distribution security bulletins and the US National Vulnerability Database (NVD)) into its database, which it updates regularly for high accuracy. This approach helps reduce false positives and ensures new CVEs are caught promptly.

The scanner maps each package in your image or system to known CVE entries and reports issues along with severity levels (e.g., LOW, MEDIUM, HIGH, CRITICAL) and available fixed versions. By generating a software bill of materials (SBOM) for the target and cross-referencing it against vulnerabilities, Trivy presents a detailed picture of risk in your software stack. Notably, Trivy is fast – the first scan might take a handful of seconds to download the latest vulnerability database, but subsequent scans are typically near-instant thanks to caching.

Beyond finding vulnerable software, Trivy can detect security misconfigurations in infrastructure definitions and container configurations. With a simple command, you can scan directories of IaC files (Terraform scripts, CloudFormation templates, Kubernetes manifests, Dockerfiles, etc.), and Trivy will automatically detect the file types and check them against built-in security policies. For example, it can catch whether a Dockerfile is using an outdated base image tag (latest instead of a specific version) or if a Kubernetes deployment manifest is missing recommended security settings.

In one case, scanning a Dockerfile revealed a medium-severity misconfiguration: The base image was set to *alpine:latest* without a fixed tag, which is a security risk because the image can change over time.

Trivy shines light on such issues with explanations (e.g., *When using a 'FROM' statement you should use a specific tag to avoid uncontrolled behavior when the image is updated*) and provides references to remediation guidance (e.g., links to the Aqua vulnerability database for misconfigurations). These checks help teams enforce best practices, from cloud environment settings to container security. The tool uses Open Policy Agent (OPA) under the hood with a library of Rego policies for common misconfigurations, and it even allows writing custom checks to tailor to your organization's standards, if needed.

Trivy also includes a scanner for sensitive information such as API keys, credentials, tokens, and other secrets that might be accidentally embedded in code or config files. When enabled, the secrets scanner will comb through file content in repositories or filesystems looking for patterns that match things like AWS secret keys, database connection strings, or private keys. The secrets scanner uses a combination of regex patterns and entropy analysis to detect high-entropy strings that likely represent secrets. Along with vulnerability and configuration issues, exposed secrets are listed in Trivy's output, making it a one-stop tool for a wide range of security checks.

As a bonus, Trivy can also identify software licenses in dependencies and flag license compliance issues (which is useful for organizations concerned about copyleft or proprietary licenses in their open source components).

Additionally, Trivy can generate SBOMs for artifacts in standard formats (CycloneDX or SPDX), resulting in an inventory of components. This component is increasingly important for supply chain security and compliance. Trivy's recent versions even allow it to scan existing SBOM files for vulnerabilities, meaning if you have an SBOM (from Trivy or another tool), you can feed it to `trivy sbom` to identify any CVEs within it.

Under the Hood

Understanding how Trivy performs these scans helps in configuring and optimizing it effectively. Trivy uses a collection of scanners specialized for different purposes (vulnerabilities, configuration issues, secrets), and it supports multiple targets or artifact types to scan. When you run Trivy, you specify a target type as a subcommand (e.g., `trivy image`, `trivy fs`, `trivy config`, `trivy repo`, or `trivy k8s`) along with the subject (an image name, a path, a repository URL, or a cluster context). Trivy auto-detects what needs to be scanned and applies the relevant scanners.

For vulnerability scanning, Trivy maintains an internal vulnerability database (Trivy DB). The first time you run it, Trivy downloads this database, which includes aggregated CVE data from many sources (distro feeds for Linux packages, language-specific vulnerability databases). This initial download is on the order of a few tens of megabytes and is what makes the first scan take a bit longer (several seconds). After that, the database is cached locally, and Trivy updates it only if it's out of date. This design frees users from manually managing vulnerability feeds.

Trivy is largely stateless and automatically keeps its data current, avoiding the heavy setup of maintaining a separate database server. The scanning itself involves unpacking the target artifact: For a container image, Trivy will either pull it from the registry (if not available locally) or use the local `docker` or `containerd` daemon image if present, then analyze the filesystem layers to identify packages and libraries.

Trivy identifies operating system packages by package manager metadata (`dpkg`, `rpm`, `apk` indexes) and language libraries with manifest files such as `pom.xml`, `package-lock.json`, `Gemfile.lock`, and so on. Identified components are matched against the vulnerability database, and any findings are reported with details such as CVE IDs, severity, and fix versions. The scanning is performed efficiently in memory, and results can be output in various formats (table, JSON, SARIF, CycloneDX SBOM, etc.) depending on your needs.

For secrets detection, Trivy's mechanism runs patterns and entropy checks across text content. It's optimized to reduce false positives by ignoring common false patterns and using context (for instance, it knows to look for an `AWS_SECRET_ACCESS_KEY=` label near a high-entropy string to confirm it's likely a key). It's not foolproof, but it's a valuable net for catching mistakes. Secrets scanning can be resource-intensive (reading lots of files), so by default it's off for image and filesystem scans.

An important internal concept is caching. Trivy caches several things to improve performance: the vulnerability database (as mentioned) and the results of recent scans. It will cache image layer analysis so that if you scan the same image tag again, it doesn't need to redo all computations (unless you force refresh). You can explicitly clear the cache, for example, if you want to ensure a full fresh scan (perhaps after updating the database or Trivy version). The cache directory is configurable and can be moved to a shared path if you have multiple users or CI agents on the same host. In advanced setups, Trivy supports a Redis cache backend, which is particularly useful in client-server mode.

Optimizing Trivy

Trivy caches the vulnerability database and scan results to accelerate subsequent runs. By default, the database will update periodically, but if you are running many scans in a short time (say, in a CI environment with many parallel jobs), you can preload or host the database internally. One approach is to run

```
trivy image --download-db-only
```

on a schedule to fetch the latest database and share the `~/.cache/trivy` directory among jobs.

In air-gapped environments with no Internet access, you can download (Trivy provides an offline database archive) and point to the database file for Trivy to use, ensuring scanning doesn't stall waiting for updates. Also, consider using the Trivy server mode: A central Trivy server can keep the database updated and serve all scanners, so each build agent doesn't redundantly download the same data.

Trivy will by default traverse through directories or image filesystems looking for all relevant files. In some cases, you might want to exclude certain paths to speed up scanning or avoid noise. Skipping large, irrelevant directories can save time. Likewise, if some files are known always to trigger false positives, those can be skipped. By tuning what Trivy looks at, you optimize runtime and focus on genuine risks.

Another performance consideration is deciding which scanners to enable. Running all scanners (`vuln`, `config`, `secret`) provides the most coverage but also does more work. If, for example, you only care about vulnerability scanning in a certain pipeline, you can stick to the default `vuln`-only scan, which will be faster than enabling everything.

Conversely, in a security audit job, you might run the full suite. Trivy gives flexibility to run the right level of scanning for context. Additionally, you can reduce output clutter by using severity filters or ignoring unfixed/unpatchable issues so that reports remain actionable. Although these don't significantly change scan time, they improve signal-to-noise, which indirectly improves efficiency by focusing attention on what matters.

If you need to scan a very large number of artifacts (like a registry with thousands of images), you can parallelize Trivy because it is just a command-line interface tool – run multiple Trivy processes in parallel, each on different targets. The main bottleneck might be the CPU and network for pulling images. Ensure the system has enough of these resources (Trivy can benefit from multiple cores, especially when scanning large images or lots of files, because it can parallelize some operations).

For container registry scanning at scale, you might script a loop that lists images and spawn concurrent Trivy scans. In client-server mode, a beefy server can help distribute the load as well. In Kubernetes, Trivy Operator automates this process by spinning up scan jobs for each image in the cluster in parallel, which is efficient.

Conclusion

Trivy's rise to prominence in the IT security world is well-deserved. It addresses a pressing need for comprehensive, continuous security scanning that keeps up with today's fast-paced development and deployment practices. ■

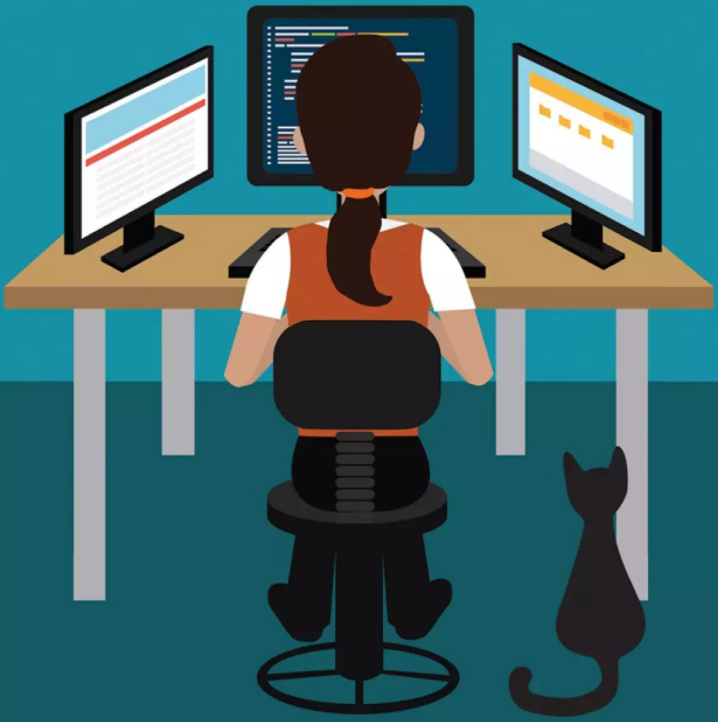
Info

[1] Trivy: [https://trivy.dev/]

[2] GitHub: [https://github.com/aquasecurity/trivy]

The Author

Marcin is a Microsoft MVP, Microsoft Certified Trainer, and conference speaker. He works as a DevOps architect and has extensive experience in Azure technologies. Visit his blog at [https://marcingastol.com/].



Want to work from anywhere?

Find remote jobs now!

OpenSource
JOB HUB

opensourcejobhub.com

@opensourcejobhub.com

Open Source JobHub

Open Source JobHub

@osjobhub@fosstodon.org

Bare metal lifecycle management with six open source tools

Tin Boxes

Many proprietary tools on the market are designed to automate server management in modern data centers, but we look at the tools offered by the open source community. By Martin Loschwitz

Installing a batch of new machines is easier said than done: It takes quite a while until all the new boxes are installed in a rack and properly wired up. Add to this the tedious task of installing an operating system on the new servers, and your on-site time increases significantly.

In many places, organizations have switched to having the systems transported to the office first, where they are installed before wending their way to the data center. However, this often logistical challenge causes considerable delays in the process. Once the server is installed on site, reinstalling it in the office is no longer an option. You either have to make your way back to the data center or cobble together an *ad hoc* installation environment based on Preboot Execution Environment (PXE) boot, Dynamic Host Configuration Protocol (DHCP), and Trivial File Transfer Protocol (TFTP) – assuming the network configuration allows you to do so.

Organizations that deal with large numbers of servers instead need a largely automated approach where pressing the power button after installing a machine in the rack is all it takes to start the initial software installation. Many commercial tools are available that can achieve this automation, including Dell's Bare Metal Orchestrator and HPE GreenLake.

Unsurprisingly, the major server manufacturers have these tools in their portfolios, but if you are looking for an open source alternative under a free license, the options quickly dry up, especially for organizations that do not want to be dependent on a commercial provider for long-term server lifecycle management. For this reason, I look at popular tools from the open source community that can be used to manage servers comprehensively at the hardware level.

Foreman, the Eternal Classic

Anyone who has spent a few years in the Linux world will inevitably have heard of Foreman [1]. The software is a classic utility in bare metal lifecycle management and has its roots in the early years of the Linux automation movement. For a long time, Foreman was so closely linked to Puppet that it didn't make sense to use the tool for anything other than setups that used that automation platform,

Photo by Beth Macdonald on Unsplash

which is still widely deployed today. These strict ties are now a thing of the past, but Foreman can act seamlessly as an external node classifier (ENC) for Puppet. If you do rely on Puppet in your environment and need software tailored to it for lifecycle management, Foreman is a good choice.

Foreman offers everything you would expect from a modern product of this type. The tool comes with helpers for data center infrastructure management (DCIM) and links the program to the out-of-band interfaces of common servers. In everyday use, the somewhat outdated IP address management (IPAM) or the significantly more modern Redfish dominate the market, although Redfish is by no means available on all modern machines. However, IPAM has no useful offering.

What's almost worse, although Foreman offers a plugin interface, no clear favorite has emerged from the community, which can be a real show-stopper in everyday use: For example, if you use NetBox, currently probably the best open source IPAM software on the market besides Foreman, you will naturally want to connect the two. Strangely, it is only possible with some additional software, such as the `smart_proxy_ipam` plugin, which acts as a kind of translator between common IPAM tools and Foreman. Native integration is not available for NetBox or other common IPAM tools.

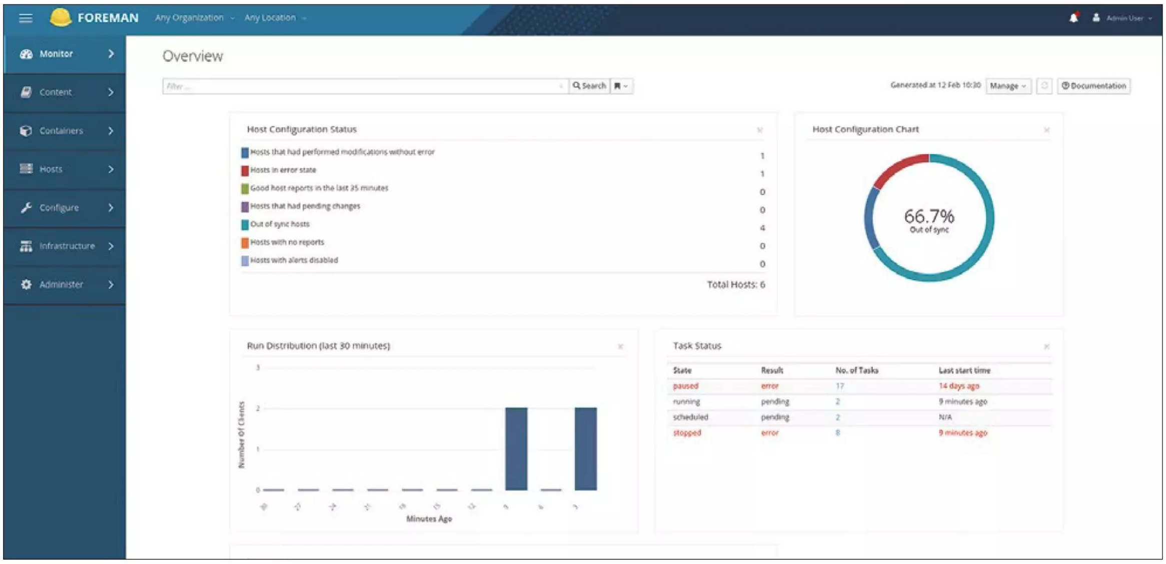
In general, you can see that Foreman fully embraces its role in an IT setup as a "single source of infrastructure truth" **[2]** and does not tolerate additional tools alongside, not because it actively prevents their use, but because it simply cannot be meaningfully connected to them. If you do decide to use Foreman, it's all or nothing: Either you totally commit yourself to the software or you avoid it. Missing essential components such as IPAM is a significant drawback, but if you can do without IPAM or manage it in parallel with a tool such as NetBox without Foreman integration, you will find Foreman to be a reliable partner that can help you manage servers effectively in most cases (**Figure 1**).

That said, the initial installation of Foreman and integration with existing systems is not exactly trivial. It usually takes a while for the software to run smoothly, and you would be well advised to plan the Foreman setup in advance. In any case, the ideal time for implementation is at the start of a new setup phase in the data center. Foreman can be installed retrospectively, but this is even more time consuming.

Tinkerbell: Success with Kubernetes

Tinkerbell **[3]** has made a name for itself in recent years. The tool was originally developed by a company named Packet, which has since been acquired by data center provider Equinix. Tinkerbell was Packet's standard tool for bare metal as a service. The tool enabled the provider's customers to create complete physical servers in a standardized API-based approach, just like AWS and the like do for virtual instances. To avoid having to completely reinvent the wheel, Packet recycled parts of the Kubernetes API for its own purposes. As a result, active use of Tinkerbell today has a lot in common with Kubernetes; in any case, it is strictly API-driven, meaning that all central work steps are triggered by calls that you send to a central API.

Under the hood, Tinkerbell has a total of five services: (1) Tink is both the central intelligence (controller) and scheduler (workflow engine) and pulls all the strings of the other components together. (2) Smee dynamically provisions DHCP and iPXE servers, (3) Hegel serves as the central API for all required metadata, and (4) Rufio works with the PBnJ remote procedure call (gRPC) server for baseboard management



Host Configuration Status

Hosts that had performed modifications without error	1
Hosts in error state	1
Good host reports in the last 35 minutes	0
Hosts that had pending changes	0
Out of sync hosts	4
Hosts with no reports	0
Hosts with alerts disabled	0

Total Hosts: 6

Task Status

State	Result	No. of Tasks	Last start time
paused	error	17	14 days ago
running	pending	2	9 minutes ago
scheduled	pending	2	N/A
stopped	error	8	9 minutes ago

Figure 1: Foreman is the classic versatile but complex tool for server lifecycle management that falls short in areas such as IPAM.

controller (BMC) interaction in the background to handle communication with BMC interfaces. Here, too, the focus is on Intelligent Platform Management Interface (IPMI). Finally, (5) Hook provides the environment from which the requested operating system is installed on dynamically provisioned systems.

Interestingly, Tinkerbell remains at the core of Equinix Metal to this day, even though the software's inventor has largely stepped away from its development. One man's loss is another man's gain: AWS now calls the shots at Tinkerbell, and it's safe to assume that the tool will also play a central role in the deployment of bare metal instances in AWS. Officially, however, the project is now under the auspices of the Cloud Native Computing Foundation (CNCF).

All told, Tinkerbell proves to be very powerful, and the fact that `kubectl` can be used to provision servers will be particularly appealing to administrators with Kubernetes experience. However, a full-fledged Tinkerbell installation is by no means a weekend project: Much preparatory work and brainpower is required before all Tinkerbell components are perfectly coordinated.

What's more, if you're expecting colorful screens and practical GUIs, Tinkerbell is definitely not for you, although it would be entirely possible to build a GUI with the existing APIs. Projects such as OpenStack work on similar principles and have been successful. A Tinkerbell GUI named Tink-Wizard has been drafted; however, this tool never made it past the early stages of development, and its GitLab directory has been abandoned for more than four years. The same author has started work on something like a terminal-based successor named Tinker, but nothing noteworthy has happened since 2022.

If you are looking for a general platform for bare metal lifecycle management, Tinkerbell is probably not the right place to go. However, if you are looking for technical underpinnings on which to offer bare metal as a service, you will find the right specifications here, because the tool was originally designed for this purpose and ticks all the boxes.

Cobbler Minimal Approach

Cobbler **[4]** is specifically aimed at admins who need basic lifecycle management features without too much fuss. It is often mentioned in the context of Foreman, but the two tools have no logical or technical dependency. Cobbler's developers make this clear, describing the tool as a Linux installation server that supports the rapid creation of an environment for installing systems on the basis of typical network technologies such as DHCP and PXE. However, Cobbler has some new tricks: It can also provide *ad hoc* DNS services, orchestration, and central storage of configuration files for systems.

Cobbler distinguishes itself with easy installation on common Linux distributions such as RHEL or compatible systems, as well as on Debian, Ubuntu, SUSE, and others by installing the `cobbler` package. The `/etc/cobbler/settings.yaml` central configuration file stores all the key parameters. Cobbler then does most of the work on the command line from new hosts to boot directly into the installation process over their BMC interfaces, a wide range of which Cobbler supports, including IPMI, Dell Remote Access Controller (DRAC), HP Integrated Lights-Out (ILO), and IBM Rational Software Architect (RSA).

The Cobbler developers cleverly worked around a problem that almost every bare metal approach has – that is, dynamic rebooting into an installation environment to roll out the desired operating system. Cobbler assumes that machines will be given a machine image at startup that is capable of triggering all other required steps. To do this, the tool works with profiles under the hood. Each profile contains all the basic settings of a system. If a profile specifies that a system should boot from the local disk, the rudimentary bootloader that Cobbler sends over TFTP immediately passes the baton to the local data carrier in the system. If the profile for reinstalling over the network is selected instead, the tool hands the task over to the network.

Much like Tinkerbell, Cobbler differs from Foreman primarily in that it does not claim to have complete control over the systems it manages. For example, Cobbler does not offer patch management, which Foreman does. Integrating Cobbler into other infrastructure services such as NetBox for IPAM and DCIM is just as difficult: You will have to do most of this work yourself. It helps to think of Cobbler mainly as a powerful tool for installing systems over a network (i.e., as software that covers some of the classic features of bare metal lifecycle management); however, Cobbler does not offer more advanced functions such as patch management, vulnerability detection, or performance evaluations.

OpenStack-Based Bifrost

Bifrost **[5]** (bye-frost) also has a fairly specific area of application and is now mostly rolled out as part of an OpenStack installation with Kayobe **[6]**.

If you hear the word OpenStack and immediately feel a chill run down your spine, rest assured: What makes Bifrost special is that it is a custom version of OpenStack Ironic specifically enhanced for operation without the other OpenStack components.

This path is a no-brainer in a way: OpenStack is known to be a platform primarily for operating a private infrastructure-as-a-service (IaaS) environment. IaaS does not necessarily mean virtual machines but can also mean appropriately provisioned physical servers. Accordingly, an early impetus within the OpenStack project was to use OpenStack to manage the systems that form an OpenStack cloud.

For a while, the name TripleO (e.g., OpenStack on OpenStack) was bandied about the community, with a logical division into an overcloud and an undercloud. Bifrost adopts up this idea and makes it possible to manage bare metal systems outside of OpenStack environments.

Unfortunately, Bifrost is not a panacea either. If you decide to embark on the Bifrost journey, you will either have to set up the tool manually and painstakingly with all the required add-on components or roll it out as part of a complete Kayobe environment so that specific local images can be used to deploy the operating system. In this respect, it is by no means a guaranteed, sure-fire success story, even if setting up Bifrost with Kayobe requires significantly less OpenStack knowledge than would a complete OpenStack installation.

Moreover, bare metal management of this nature can practically only be implemented if you start with fresh hosts on a greenfield site. Integrating Bifrost into an existing platform at a later stage is virtually impossible – not that it is easy with other tools, but it can be done far faster than with Bifrost.

Red Hat Satellite

In an article on bare metal lifecycle management, you just cannot ignore the industry leader, Red Hat Satellite [7] (Figure 2). The vendor markets the software as a jack of all trades: Satellite takes care of provisioning the systems after installation in the rack, as well as subsequently fueling them with the desired software (e.g., Ansible).

Satellite also implicitly manages the subscriptions activated for each host. According to the vendor, you no longer need to worry about outdated software and vulnerabilities, because the tool takes care of software management and, in particular, the installation of patches – even completely autonomously if you so desire. Red Hat has also done a good job with the setup: Satellite is quick and easy to install, thanks in no small part to the detailed documentation.

The downside is that Satellite is primarily aimed at environments where RHEL is the predominant operating system. Although some other systems can also be managed with the product, full support from the manufacturer is primarily available for Red Hat-affiliated installations. Either way, the tool comes at a price: Satellite is one of the few tools in the test that requires a subscription, which means that your organization will have to dig deep into their pockets.

Canonical MAAS

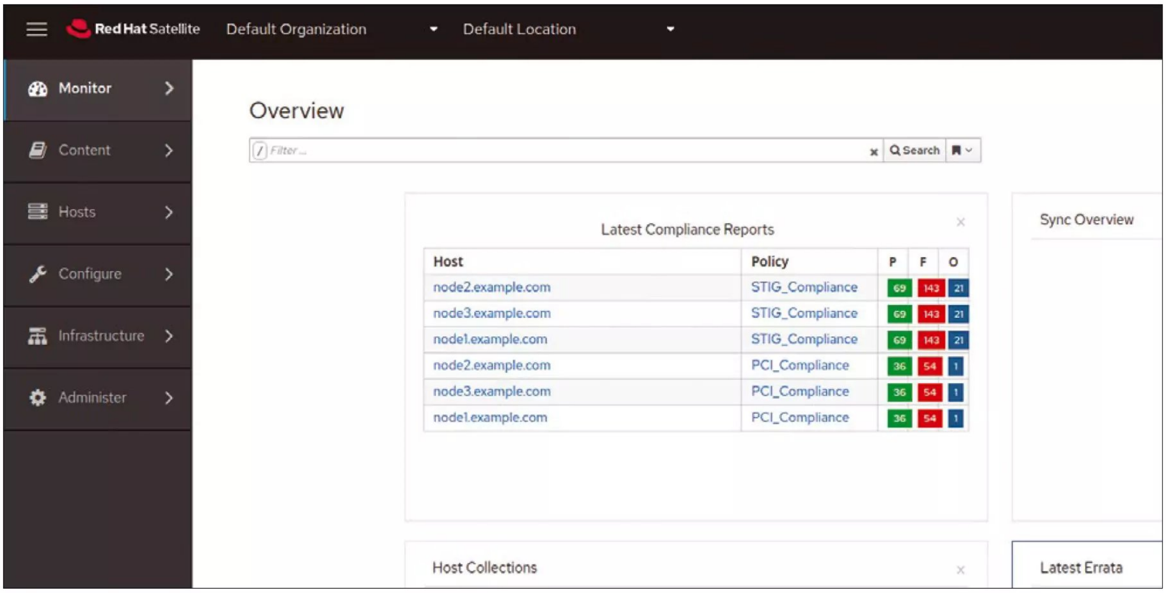
The same applies to Canonical MAAS (a metal-as-a-service server orchestration tool) [8], which is primarily aimed at Ubuntu customers, promising them seamless management of physical systems (Figure 3).

The provider explicitly points out that MAAS can also be used to create non-Ubuntu installations on systems such as RHEL, VMware ESXi, and even Windows. The variety of supported hardware is also extensive.

MAAS installation is refreshingly straightforward with the Snap package format. Installation is followed by various configuration steps, such as activating the Transport Layer Security protocol (TLS) or enabling communication between MAAS and Juju, Canonical's automation tool. You then integrate your physical systems by reconfiguring them to PXE boot and downloading a boot image from MAAS for the first boot. MAAS then adds the machines to its database.

Additionally, MAAS can be connected directly to a server's BMC interface via power drivers, primarily with IPMI or Redfish over IPMI. After that, machine deployment can start, to which, true to its name, MAAS functionality is limited.

If you want advanced lifecycle management, you will need Ubuntu Landscape and a matching Ubuntu Pro subscription. Landscape then enables seamless administration of your systems. If you don't want to spend money on Ubuntu Pro, you have the option, as with Tinkerbell or Cobbler, of configuring your



Latest Compliance Reports

Host	Policy	P	F	O
node2.example.com	STIG_Compliance	69	143	21
node3.example.com	STIG_Compliance	69	143	21
node1.example.com	STIG_Compliance	69	143	21
node2.example.com	PCI_Compliance	36	54	1
node3.example.com	PCI_Compliance	36	54	1
node1.example.com	PCI_Compliance	36	54	1

Figure 2: Satellite is considered the ideal solution for bare metal lifecycle management in Red Hat-style setups.

own automation (e.g., on the basis of Ansible) so that it takes systems under its wing and manages them immediately after the operating system has installed. If you combine this arrangement with AWX, you even get a nice GUI to facilitate operation.

Conclusions

Bare metal lifecycle management with open source software remains a complex topic. Unlike proprietary tools from major hardware manufacturers, tools from the free/libre open source software (F/LOSS) universe offer the fundamental advantage of not locking users into a vendor and not creating financial dependencies. To put it bluntly, an organization that designs its own bare metal lifecycle management from free components is investing in its independence.

That said, no single piece of free software on the market is so universal that it perfectly covers all everyday requirements. Foreman comes closest, but falls short when it comes to modern features such as integrating IPAM systems like NetBox. Tinkerbell is one of the more advanced offerings, but it relies on a specific deployment setup and alienates administrators who are not comfortable with the Kubernetes approach to problem solving. Satellite can do a great deal, but primarily targets Red Hat customers. The same applies to MAAS from Canonical and its dependency on Ubuntu.

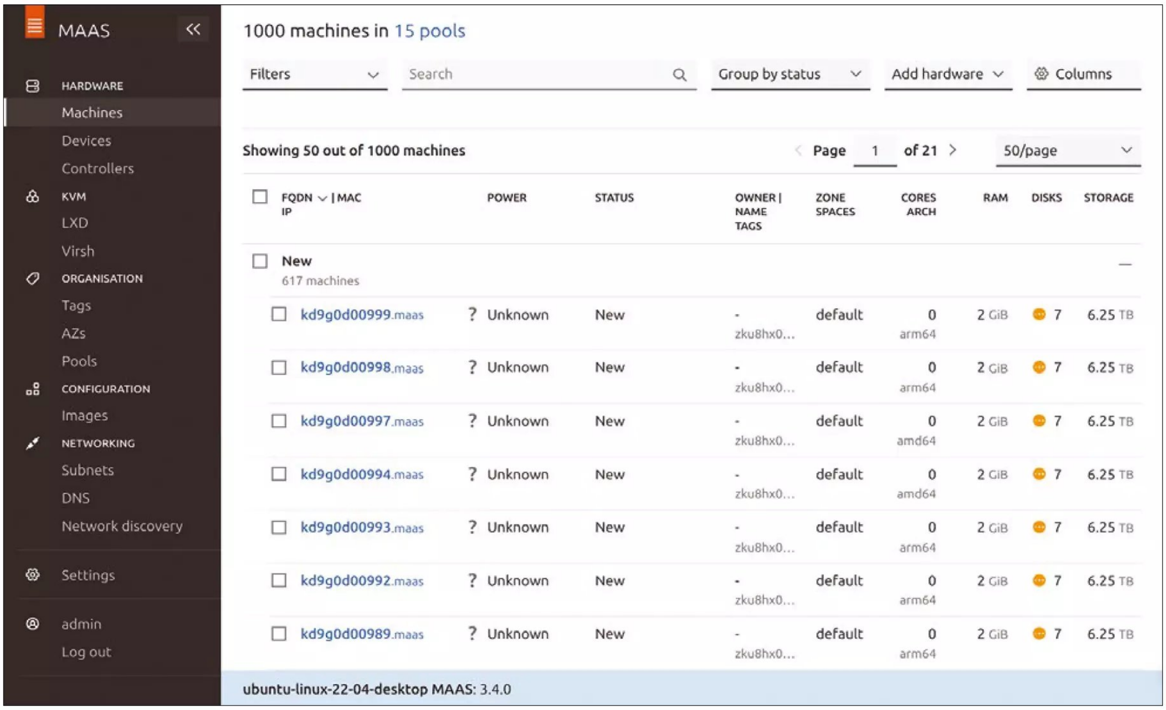
Bifrost with Kayobe is a niche product, and Cobbler can be adapted to most environments if you are not afraid to put in the manual work required. Administrators who need lifecycle management would be well advised to document their requirements in detail and thoroughly test the available free alternatives on the market to find the ideal tool for their specific use case. ■

Info

[1] Foreman: [https://theforeman.org/]
[2] Single source of truth: [https://www.theforeman.org/introduction.html]
[3] Tinkerbell: [https://tinkerbell.org/]
[4] Cobbler: [https://cobbler.github.io/]
[5] Bifrost: [https://opendev.org/openstack/bifrost]
[6] Kayobe: [https://www.openstack.org/software/releases/dalmatian/components/kayobe]
[7] Red Hat Satellite: [https://www.redhat.com/en/technologies/management/satellite]
[8] Canonical MAAS: [https://maas.io/]

The Author

Martin Loschwitz is the founder and managing director of True West IT Services GmbH, which offers scalable IT infrastructure based on OpenStack and Kubernetes.



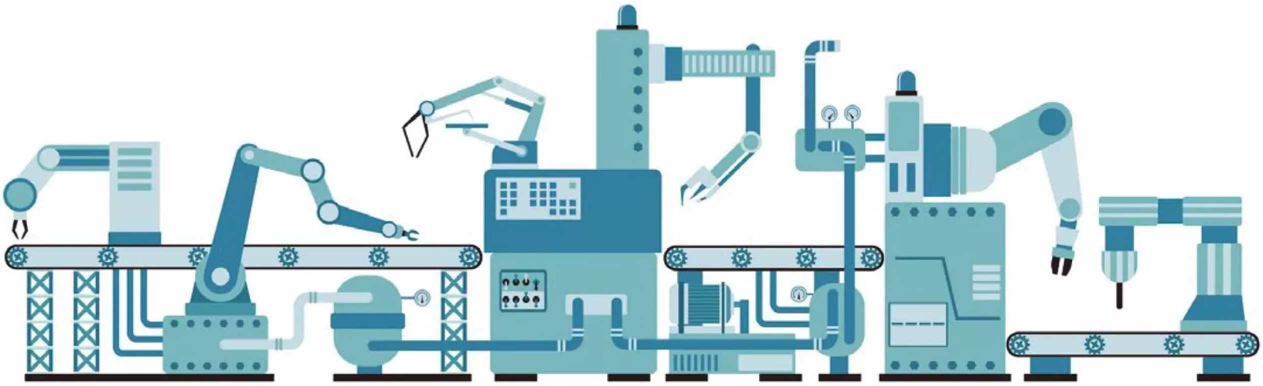
1000 machines in 15 pools

Showing 50 out of 1000 machines

☐ FQDN ∨ | MAC IP	POWER	STATUS	OWNER | NAME TAGS	ZONE SPACES	CORES ARCH	RAM	DISKS	STORAGE
☐ New 617 machines								
☐ kd9g0d00999.maas	? Unknown	New	- zku8hx0...	default	0 arm64	2 GiB	7	6.25 TB
☐ kd9g0d00998.maas	? Unknown	New	- zku8hx0...	default	0 arm64	2 GiB	7	6.25 TB
☐ kd9g0d00997.maas	? Unknown	New	- zku8hx0...	default	0 amd64	2 GiB	7	6.25 TB
☐ kd9g0d00994.maas	? Unknown	New	- zku8hx0...	default	0 amd64	2 GiB	7	6.25 TB
☐ kd9g0d00993.maas	? Unknown	New	- zku8hx0...	default	0 arm64	2 GiB	7	6.25 TB
☐ kd9g0d00992.maas	? Unknown	New	- zku8hx0...	default	0 arm64	2 GiB	7	6.25 TB
☐ kd9g0d00989.maas	? Unknown	New	- zku8hx0...	default	0 arm64	2 GiB	7	6.25 TB

ubuntu-linux-22-04-desktop MAAS: 3.4.0

Figure 3: MAAS is Canonical's standard tool for managing physical systems, so you can deploy the operating system and afterward manage the servers.



Monitoring IoT devices

In the Machine Shop

Keeping track of a large number of Internet of Things devices means understanding IoT networks, developing suitable monitoring metrics, and (ideally) integrating the IoT with infrastructure monitoring. By Lothar Burger

When IT teams set out to monitor the Internet of Things (IoT), they too often settle for situations where, say, a networked sensor sends plausible measurements at an expected interval, which unfortunately means that admins could be overlooking critical scenarios. For example, the device could be repeatedly rebooting, which could mean a hardware failure in the making or that the battery charge is dropping rapidly because of the numerous reboots. Finally, operations managers could miss a device's unusual communication volume, indicating that it is not just communicating with the application in the back end as intended but is also carrying out attacks against external websites as part of a botnet.

The IT world at the edge and in the back end is networked over IP and can be monitored with the familiar methods and tools of classic IT monitoring, which is why I focus in this article on monitoring devices and their connectivity. However, IT and IoT do not need to be separate monitoring silos; instead, consolidating the monitoring data from both worlds on an observability platform makes sense to achieve a holistic view of the IoT system and its components.

Lead Image © alexutemov, 123rf.com

Understanding IoT Networks

Figure 1 shows the typical structure of an IoT system comprising three layers: devices, the edge, and the back end. The layers are connected by networks. The applications on the devices communicate with applications in the back end or on edge servers. Edge servers enable the processing of IoT data and storage of the data close to its place of generation. Devices can be connected either by cable or wirelessly. Cable-based variants include fieldbus systems (KNX, Modbus, Profibus, etc.) and Ethernet-based systems (EtherCAT, EtherNet/IP, single-pair Ethernet, etc.). However, admins often prefer wireless infrastructures – either because the objects that need to be networked are mobile (e.g., vehicles, load carriers, or mobile tools), or because wireless networks are easier, faster, and cheaper to implement.

Wireless technologies such as EnOcean, Zigbee, Z-Wave, Bluetooth LE, and Bluetooth Mesh are used in building automation, whereas technologies capable of bridging longer distances, such as LoRa and LoRaWAN, NarrowBand IoT (NB-IoT), and mesh networks that are based on IEEE 802.15.4, are used in smart city applications. For networks facing the back end and within the edge, you tend to see IP-based networks whose communication media are well known from legacy IT: Ethernet, WiFi, mobile communications, Internet VPN, software-defined (SD)-WAN, and so on.

IP routers or gateways act as bridges between the networks with the devices and those of the edge or back end, depending on whether the end devices support IP. The trend toward all-IP communication is also evident in IoT devices, either natively or with the help of a special adaptation layer – see also the Internet Engineering Task Force (IETF) requests for comment (RFCs) and drafts on IPv6 over low-power wireless personal area networks (6LoWPAN) and IPv6 over constrained node networks (6lo) [1]. That said, narrowband wireless networks are rarely suitable for the overhead associated with IP. In this case, gateways need to handle the required protocol conversions. Some IoT systems do not require an edge at all, such as telematics systems in which the on-board units installed in vehicles are connected directly to the back end by mobile communications and Internet-based VPN.

Challenges in IoT Monitoring

Monitoring IoT devices and their connectivity differs from classic IT infrastructure monitoring for several reasons and presents new challenges for ops teams. First and foremost is the equipment of the IoT devices and what is often a narrowband connection. Limited processing and storage capacities rule out installing monitoring agents or frequent status polling by a central monitoring server. If the devices are also only allowed to transmit a few bytes per hour, the monitoring data would compete with application data on the network. Also, you cannot assume that IoT technology will always be ready to receive status queries, especially for battery-powered devices that need to keep uptimes as short as possible to ensure a long battery life.

Industrial IoT in particular is characterized by a heterogeneous landscape of IoT devices: sensors, actuators, mobile tools, indoor positioning systems, cameras, and legacy systems that have been retrofitted to be IoT-compatible. This heterogeneity is accompanied by a variety of hardware configurations, operating systems, communication technologies, and protocols.

Developing Metrics

Returning to Figure 1, you can already see the starting points for monitoring IoT devices and their connectivity: the IoT application's data stream, the device network, and the IoT devices. You can at least keep an eye on devices that do not support monitoring functions by paying attention to operation-critical events or characteristics in the IoT application's data stream in the back end or at the edge:

- Devices connecting to the edge or back end
- Failed connection attempts by the device (possibly differentiated by cause)
- Devices sending a plausible application message
- Devices sending an implausible application message (possibly differentiated by error type)
- Application messages successfully reaching the device
- Application messages failing to reach the device

You can use this list to derive metrics such as the last successful connect time, the successful connection rate, the failed connection rate, the message rates in the send and receive directions, and the number of bytes sent or received. These metrics can be computed not only for individual devices, but also for groups of devices or the entire fleet. In combination with threshold values, the right choice of

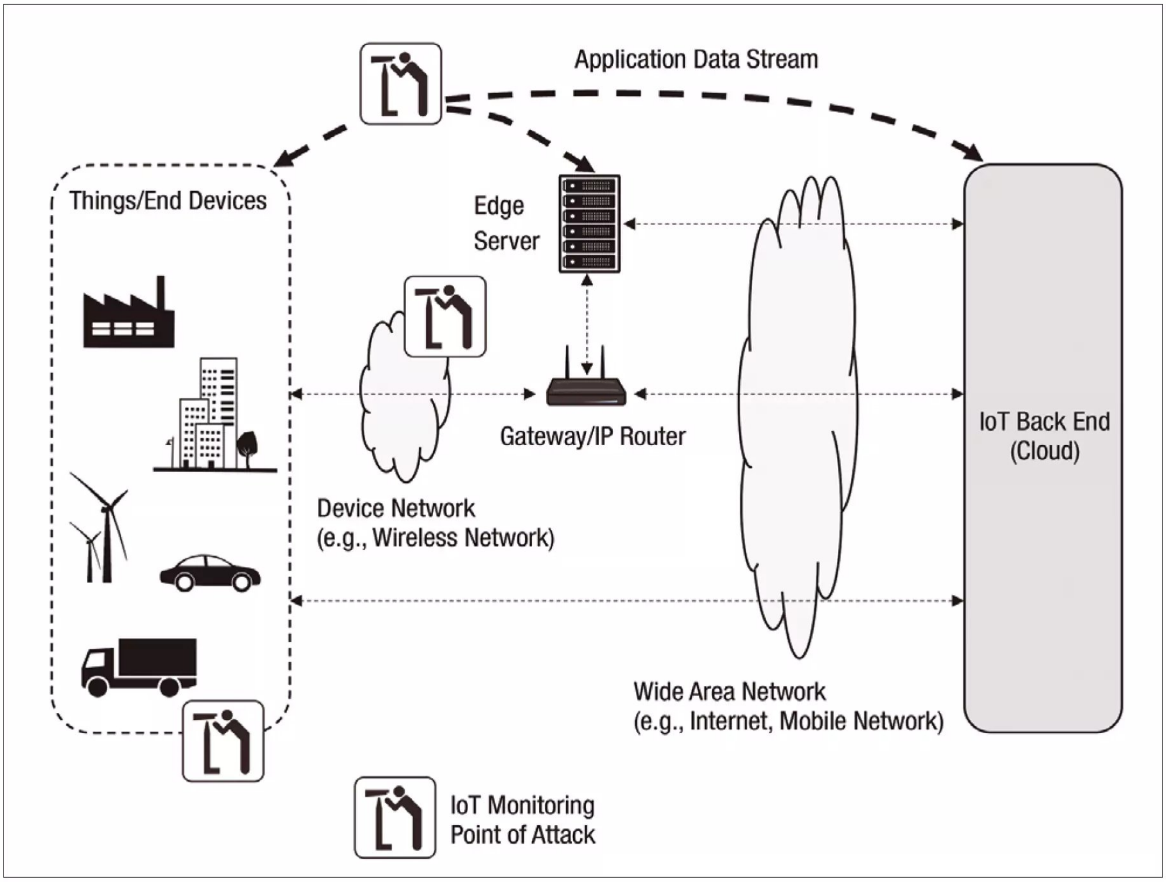

Application Data Stream
Things/End Devices
Edge Server
Gateway/IP Router
Device Network (e.g., Wireless Network)
Wide Area Network (e.g., Internet, Mobile Network)
IoT Back End (Cloud)
IoT Monitoring Point of Attack

Figure 1: Data streams, device networks, and IoT devices are the starting points for monitoring.

tool will be able to raise the alert if values are exceeded or not reached. The application data stream can also be monitored to detect anomalies. For example, large telematics systems often show typical, recurring daily patterns in the message rate. Faults affecting a large number of end devices manifest themselves as dips in the expected curve, which can be detected easily by statistical methods. The magnitude of the deviation can also be used to estimate how many devices are affected or the size of the geographical region affected by the disruption. Anomalies in IoT systems sometimes become apparent to those responsible even without AI, but only through experience and the use of statistics tools.

Example: Microsoft Azure IoT

Monitoring the application data stream is an efficient way to detect and alert for basic error states in an IoT landscape. Unsurprisingly, IoT platforms such as Microsoft Azure IoT offer features to match. The overview in Figure 2 shows how IoT Hub [2] and Azure Monitor [3] services monitor the IoT data stream via the alert path. Azure Monitor also offers additional features for analysis, visualization, and automatic actions, which I will not be discussing in detail here.

IoT Hub is the central component that handles communication with devices. In addition to numerous other metrics, the service also collects the following device-related metrics:

- Number of devices connected to IoT Hub
- Overview of cloud-to-device messages whose transmissions were aborted
- Cloud-to-device messages that were successfully accepted by the device
- Number of rejected cloud-to-device messages
- Device-to-cloud messages successfully received

These metrics (aka “platform metrics” in Azure) are typically measured every minute and forwarded to Azure Monitor. Platform metrics are collected and forwarded automatically; no explicit configuration is required.

In Azure Monitor, alert rules can trigger alerts when a metric exceeds or drops below a threshold value.

However, platform metrics do not cover all aspects required for comprehensive monitoring. In a security-critical IoT system, for example, the current rate of connection attempts by unknown devices could be of interest. The same applies to the rate of connection requests that have failed because of certificate errors. These events can be logged on IoT Hub and forwarded to Azure Monitor, which requires a diagnostic setting on IoT Hub that specifies the log entry categories you want to cover (e.g., connections, device telemetry, and device identity operations). Special alert rules that are linked to the log database in Azure Monitor with a search query can trigger alarms, as with metrics.

Monitoring the Device Network

One common error pattern in IoT systems is communication between the back end and individual devices or entire device groups being interrupted. Monitoring the application data stream can detect a disruption, but it cannot tell you whether the problem is due to a faulty software update or a network error. Additional monitoring of the device network is therefore essential to identify the causes of error, particularly if you use wireless networks that operate in a license-free frequency range.

These networks can be exposed to external influences that are difficult to control. For example, newly commissioned third-party systems that use the same frequency range in close proximity can cause spontaneous interference in device communication. The signal quality on the wireless link could also deteriorate gradually – for example, because one or more devices have entered the radio shadow of a building complex currently being built.

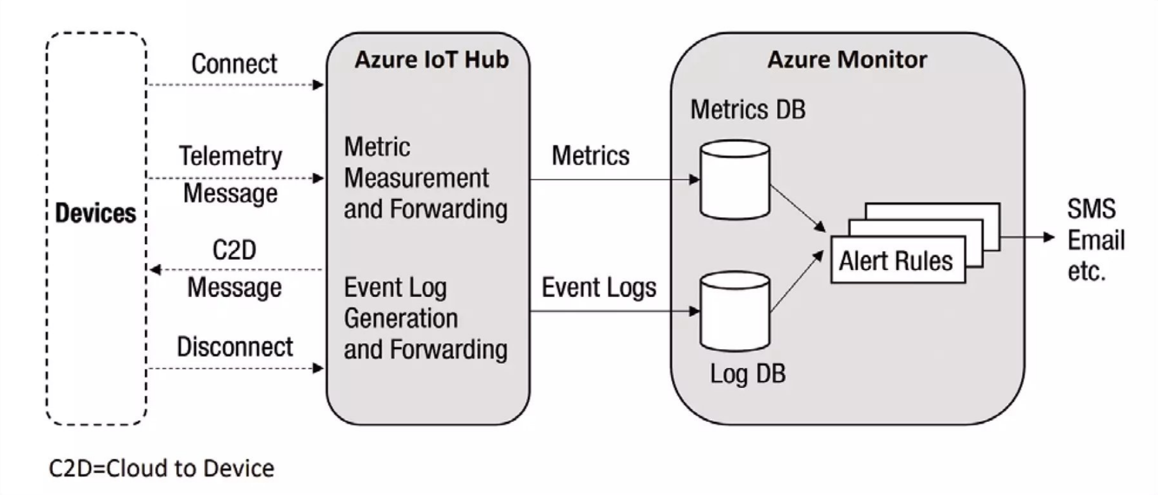


Figure 2: Monitoring in Azure IoT includes IoT Hub and Azure Monitor services.

As an example, I'll look at monitoring a device network with LoRaWAN **[4]** and a service from Amazon Web Services (AWS). LoRaWAN belongs to the low-power wide-area network (LPWAN) class; its components are devices, gateways, and three types of uplink servers (network, join, application). A proprietary wireless technology by Semtech Corporation is used between IoT devices and gateways. In North America, it uses 915MHz of the unlicensed industrial, scientific, and medical (ISM) band (902-928MHz range); Australia also uses the 915MHz band, but within the narrower 915-928MHz range. In Europe, the technology uses one of the license-free frequency bands at 868MHz, and worldwide, it uses 433MHz. Network servers route messages between gateways and the other uplink servers, and the gateways and all uplink servers are connected over an IP backbone.

The AWS IoT Core for LoRaWAN service **[5]** maps the function of network and join servers in the cloud (**Figure 3**) and thus removes the need for LoRaWAN network operators to operate their own network and join servers. This setup allows LoRaWAN gateways to connect directly to the cloud. The application server function can be mapped in the same cloud by additional AWS services. For smooth interworking between gateways and the AWS service, the LoRa Basics Station open source software package must be running on the gateways.

For every message a gateway receives from a device, the gateway evaluates the quality of the wireless link. The metrics measured for this purpose are added as metadata to each uplink message:

- Signal-to-noise ratio (SNR) measures the quality of the payload signal and indicates how well it is differentiated from background noise or interference signals.
- The received signal strength indicator (RSSI) indicates the field strength of a signal.
- The loss rate of uplink messages can be derived from jumps in the frame counter.

The network server can use these metrics in the downlink direction to determine which gateway is best for reaching an end device. Metrics of this kind are also useful for monitoring, which is why AWS makes them available on a dashboard or through an API. These datasets can be aggregated over a specified time interval, either for individual devices or for all devices served by a gateway. Mapping the data over time gives you an idea of where and how the quality of the wireless network is changing. Possible actions you can take include changing data rates, replacing gateways, positioning devices closer to a gateway, or installing additional gateways.

Monitoring Agents

For IoT end devices that have sufficient processing and storage capacity and are also directly integrated into an IP network, deploying a monitoring agent, as on other IT systems, makes sense. Doing so means that IoT and IT can be integrated into a common monitoring system. Therefore, established product providers from the IT monitoring sector unsurprisingly also have special IoT agents in their portfolios. One example is Datadog, which offers a lightweight version of its monitoring agent tailored to IoT **[6]**. This software runs on Linux and supports common hardware platforms for IoT systems (x64, ARM). According to the manufacturer, the current version of the agent requires 36MB of RAM, 63MB of memory space, and a network bandwidth of 237bps (uplink) and 79bps (downlink), which enables Datadog to collect CPU utilization, I/O rates, TCP/IP statistics, and performance data with the systemd Linux system manager. The agent also supports the collection of logs and is capable of integrating individual application metrics.

In some cases, IT needs monitoring points on devices but does not want to install commercial monitoring agents for cost reasons. If the IoT application is under your control, monitoring can also be integrated into the application. The results can then be either attached to application messages as metadata or

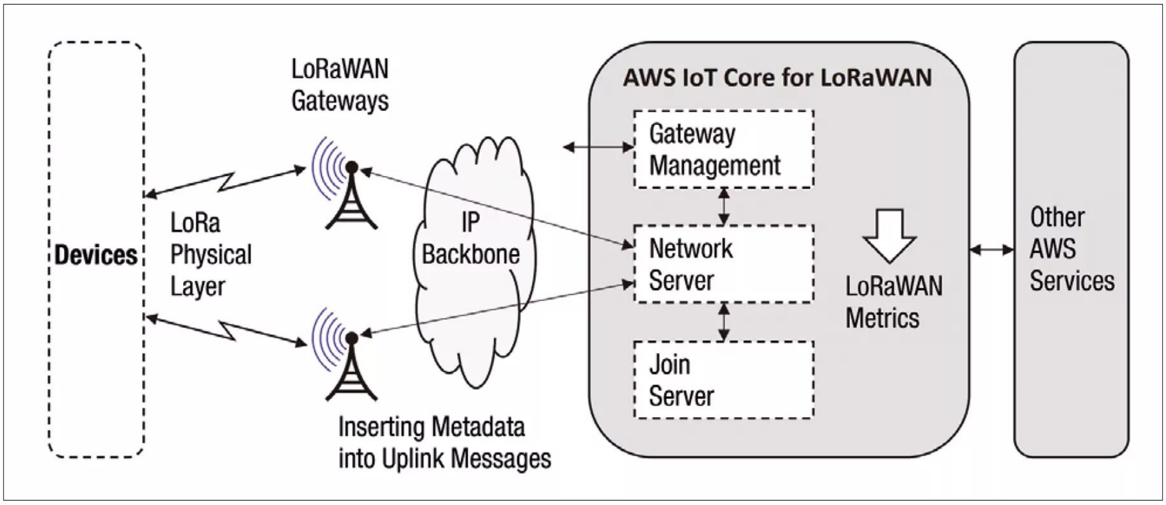


Figure 3: With AWS IoT Core, LoRaWAN gateways can connect directly to the cloud.

transmitted to the back end separately in the scope of a heartbeat message. Finally, another alternative is to rely on OpenTelemetry standards and send the monitoring data to open source software in the back end.

Conclusions

In this article, I described three potential approaches to monitoring IoT systems and investigated them with examples from industrial best practices. In many cases, monitoring is based on the application data stream. Which additional components are required depends on the specific requirements of the system. If no business-critical processes depend on the correct function of an IoT device and the device is also well secured in its network environment, it might be fine simply to keep track of the application data stream. However, IoT devices are increasingly involved in business-critical processes that are susceptible to disruption from malfunctions (e.g., monitoring goods that require refrigeration during transport and storage). A failure of the monitoring IoT infrastructure can lead not only to the loss of valuable commodities, but also to the violation of industry-specific regulations. IoT systems play a similarly critical role in manufacturing, medicine, building management, and numerous other fields of application. Therefore, IoT has no universal best choice of monitoring solution; you always need to consider whether a particular software is suitable for your field of application against the backdrop of overarching business objectives, industry-specific regulations, security risks, and costs. ■

Info

[1] IETF on 6LoWPAN and 6lo: [https://datatracker.ietf.org/wg/6lo/documents/]
[2] Azure IoT Hub: [https://learn.microsoft.com/en-us/azure/iot-hub/]
[3] Azure Monitor: [https://learn.microsoft.com/en-us/azure/azure-monitor/]
[4] LoRa Alliance homepage: [https://lora-alliance.org]
[5] AWS IoT Core for LoRaWAN: [https://aws.amazon.com/de/iot-core/lorawan/]
[6] Datadog IoT Agent: [https://docs.datadoghq.com/agent/iot/?tab=deb]

Author

Dr. Lothar Burger works as an independent IT consultant. He has professional experience in software development, telecommunications, computer networks, telematics, and the Internet of Things. He began his professional career with a degree and doctorate in computer science at the Karlsruhe Institute of Technology.

Discover the past and invest in a new year of IT solutions at Linux New Media's online store.

Want to subscribe?

Searching for that back issue you really wish you'd picked up at the newsstand?

shop.linuxnewmedia.com/shop

shop.linuxnewmedia.com/shop

DIGITAL & PRINT SUBSCRIPTIONS

SPECIAL EDITIONS

Internal central repositories for PowerShell modules

In Stock

Optimize the way you organize and maintain scripts by bundling PowerShell modules and deploying them internally. By Philip Lorenz

PowerShell has established itself as an indispensable tool for admins. Best practices for creating efficient and reusable code dictate abstracting the code to create functions that can then be bundled into modules for logical organization and to make them easily accessible to all users. Modules do not just facilitate code reuse, they also help you keep track of complex scripts, while ensuring that the code remains well structured and maintainable. The question quickly arises as to how best to store these modules and make them accessible.

Private Repository Security

The PowerShell Gallery, a public repository, provides a platform where anyone can share their modules, with the advantage of making modules available to a broad community of developers. Of course, business organizations will typically want to avoid making their own modules available to the public for various reasons (e.g., security, sensitive information, company policy). Another aspect is the protection of intellectual property; companies often do not want their proprietary content to be publicly accessible.

One proven solution in this area is to set up your own internal repository to store proprietary modules securely and make them accessible to authorized employees only and to be centrally managed, updated, and distributed in a controlled manner.

In this article, I look at the options for bundling proprietary PowerShell functions as modules and making them available to employees in a central module repository. I also explain how to create and maintain such a repository and how to integrate it into your IT department's workflow. I will look at the various steps required to set up an internal repository, including selecting the appropriate infrastructure, configuring access controls, and ensuring continuous maintenance of the repository.

Modularization Benefits

PowerShell modules are collections of related functions, cmdlets, variables, and other resources that are organized and distributed as a single unit. The main purpose of a module is to ensure code reusability and improve the structure of PowerShell scripts. By modularizing tasks from specific application domains, such as VMware, Active Directory, or Azure, you can reuse functions without having to rewrite the code.

Additional benefits of PowerShell modules include: improved code maintainability and readability from clear structuring; versioning support, which allows the use of different versions of a module; support for consistency across the organization through reusable building blocks; simplified updates and improvements, because changes only need to be made to the module itself; and enhanced security through the ability to expose only certain functions.

Basic Structure of a Module

A PowerShell module comprises several components:

- A PSD1 manifest file that contains metadata for the module, such as the version, author, and dependencies
- A PSM1 module file, where the PowerShell code is stored
- A directory structure that typically organizes modules into separate folders with at least the manifest and module files

When importing a module, PowerShell searches for the module name in the module directory (`$PSModulePath`). If a manifest is present, it is checked and the module's functions and cmdlets are loaded into the

Photo by Kenny Eliason on Unsplash

current session. An imported module uses memory until you unload it with the `Remove-Module` command.

Developing Your Own Modules

The following example creates two simple functions that you will pack into a PowerShell module:

```
# Function 1: Outputs "Hello, World!"
function Get-HelloWorld {
Write-Output "Hello, World!"
}
# Function 2: Outputs the current date
function Get-CurrentDate {
Write-Output (Get-Date)
}
```

Each module is best saved in its own directory, with a directory name that matches the module name:

```
New-Item ➘
  -Path "$HOME\Documents\
WindowsPowerShell\➘
         Modules\MyFirstModule" ➘
  -ItemType Directory
```

Now, create the PSM1 file that will store the functions and create a manifest file (PSD1) to store additional information about the module (Listing 1). Modules can then be imported and used in the PowerShell session.

Function and Structure of Repositories

A module repository is a storage location where you centrally store and manage PowerShell modules, allowing users and administrators to install, update, and manage modules easily. Centralized deployment ensures that everyone involved uses the latest versions at all times, which, in turn, enhances the security and reliability of the scripts provided. It also simplifies collaboration within teams, because all members have access to the same modules.

Although public repositories such as the PowerShell Gallery are accessible to everyone, private repositories can store organization-specific or security-relevant modules. Organizations can then host internally developed modules that meet specific requirements or contain sensitive data. As a result, control over the provided content remains entirely in the hands of the organization. (However, passwords should be kept in a secret store and not in the code.)

PowerShell uses NuGet as its back-end protocol for module repositories. Modules can be managed by an approach that should be familiar from other package managers, such as `npm` or `pip`. NuGet offers proven, reliable methods for deploying and managing software packages. To use a repository in PowerShell, you first need to register it, which is usually done with the `Register-PSRepository` command, thus making the software directory known in the PowerShell environment and enabling access to the modules:

```
Register-PSRepository ➘
  -Name "MyRepository" ➘
  -SourceLocation ➘
   "https://<myrepo.example.com>/nuget" ➘
  -InstallationPolicy Trusted
```

The commands

```
Install-Module ➘
  -Name "ModuleName" ➘
  -Repository "MyRepository"
Update-Module -Name "ModuleName"
```

then install and manage the modules from the repository.

Sonatype Nexus

Sonatype Nexus [1] is comprehensive open source software for managing a wide variety of artifacts, including PowerShell modules. The platform is useful for deploying and storing modules and acts as a proxy for external sources, such as the PowerShell Gallery, so teams can centrally manage PowerShell modules and make sure only verified versions of dependencies are used.

Nexus supports a wide range of repository types, making it easy for companies to store and organize all important artifacts in a centralized environment. As a proxy server, it enables the efficient use of external libraries by caching content and making it reusable, offering download time reductions and other benefits. Additionally, you can define strict policies for accessing and using PowerShell modules to improve security and traceability in development processes. Nexus can be installed by Chocolatey:

```
choco install nexus-repository
```

Installing in Docker (Compose) is also possible. To do so, you need to save the following content in the `docker-compose.yaml` file:

```
services:
nexus:
image: sonatype/nexus3
volumes:
- "nexus-data:/nexus-data"
ports:
- "8081:8081"
volumes:
nexus-data: {}
```

To start the container, type:

```
docker-compose up
```

Publishing Modules in Nexus

To publish PowerShell modules in Sonatype Nexus, you first need to do some preparatory work to create a repository and configure the required settings. The first step is to create a new repository in Nexus. Navigate to the settings in the Nexus web user interface, select *Create repository* (Figure 1), enter a name for

Listing 1: Creating the PSM1 File

```
New-Item -Path "$HOME\Documents\WindowsPowerShell\Modules\MyFirstModule" -Name "MyFirstModule.psm1"
   -ItemType "File"
New-ModuleManifest -Path "$HOME\ Documents\WindowsPowerShell\Modules\MyFirstModule\MyFirstModule.psd1"
   -RootModule "MyFirstModule.psm1" -Author "Philip Lorenz" -Description "Example of a simple module"
```

the repository, and select the *nuget (hosted)* option.

Before you publish a PowerShell module, make sure *NuGet API Key Realm* is activated, which Nexus needs to generate API keys for authentication. You can enable the realm in the *Settings* | *Security* | *Realms* configuration. After doing so, generate an API key in the Nexus web interface so that you can provision the modules later (Figure 2).

After creating the repository in Nexus and generating the API key, you need to register the repository with PowerShell,

```
Register-PSRepository ➘
  -Name <Repo name in PowerShell> ➘
  -SourceLocation http://localhost:8081/➘
   repository/<Repo name in Nexus>/ ➘
  -PublishLocation http://localhost:8081/➘
   repository/<Repo name in Nexus>/ ➘
  -PackageManagementProvider nuget ➘
  -InstallationPolicy Trusted
```

taking care to replace the name of the repository in PowerShell and Nexus and the URL to suit your environment. This command adds the Nexus repository as a PowerShell repository from which you can then publish and install its modules. To publish the module to Nexus, use the `Publish-Module` command, specifying the API key you generated earlier:

```
Publish-Module ➘
  -Name ./MyFirstModule.psd1 ➘
  -Repository <PS repo name> ➘
  -Verbose -NuGetApiKey "<API key>"
```

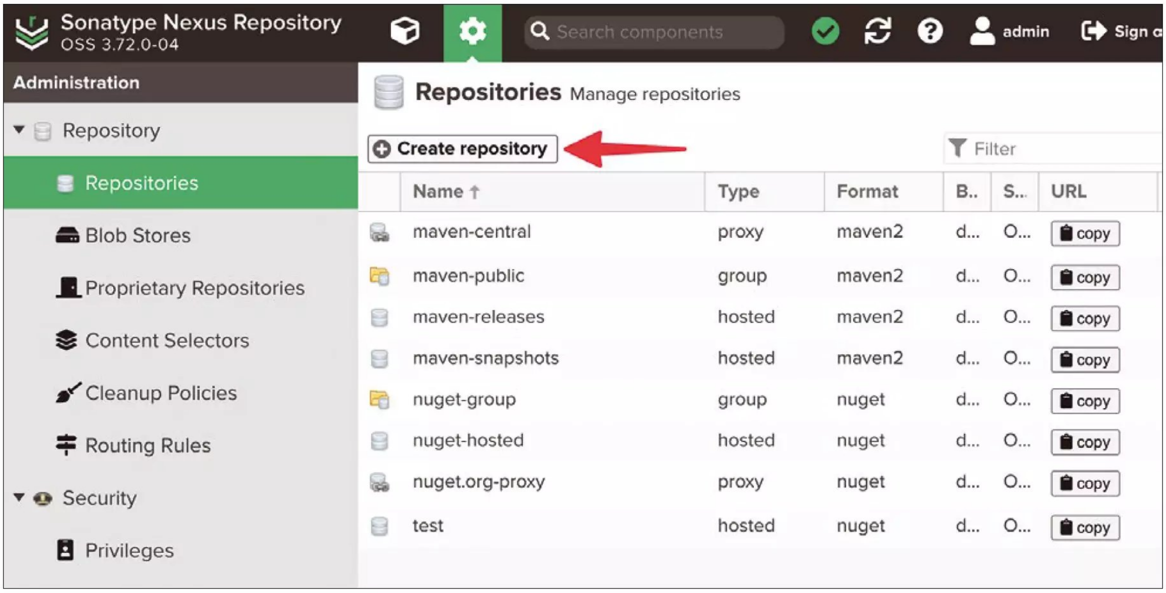
The command publishes the `MyFirstModule` module in the Nexus repository. To list the modules available in the repository, use the `Find-Module` command:

```
Find-Module -Repository <PS repo name>
```

To install the published module, use the `Install-Module` cmdlet; for example:

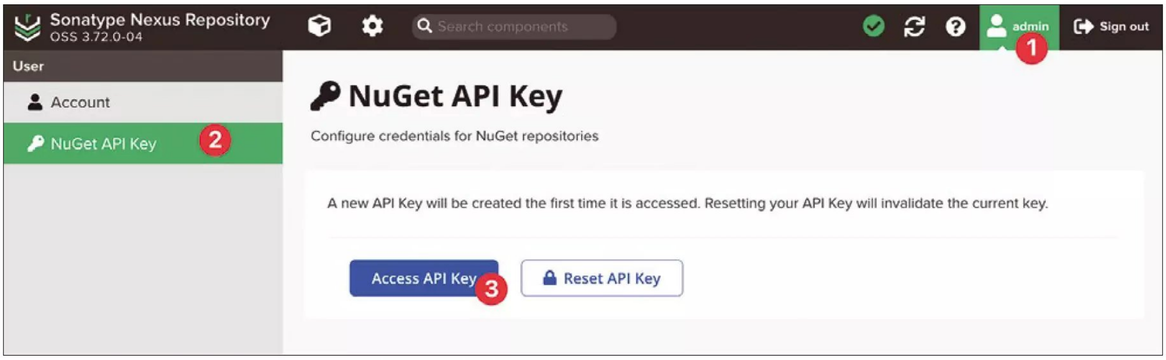
```
Install-Module ➘
  -Name MyFirstModule ➘
  -Repository "<PS repo name>" ➘
  -Force
```

The `-Force` parameter ensures that the module is updated, if needed. To publish a new version, you first need to update the module version in the `MyFirstModule.psd1` file. The module


Sonatype Nexus Repository
OSS 3.72.0-04
Search components
admin
Sign o
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Repositories Manage repositories
Create repository
Filter

Name ↑	Type	Format	B..	S...	URL
maven-central	proxy	maven2	d...	O...	copy
maven-public	group	maven2	d...	O...	copy
maven-releases	hosted	maven2	d...	O...	copy
maven-snapshots	hosted	maven2	d...	O...	copy
nuget-group	group	nuget	d...	O...	copy
nuget-hosted	hosted	nuget	d...	O...	copy
nuget.org-proxy	proxy	nuget	d...	O...	copy
test	hosted	nuget	d...	O...	copy

Figure 1: Navigate to the settings in the Nexus web interface and select *Create repository* to create a new repository.


Sonatype Nexus Repository
OSS 3.72.0-04
Search components
admin
1
Sign out
User
Account
NuGet API Key
2
NuGet API Key
Access API Key
3
Reset API Key

Configure credentials for NuGet repositories

A new API Key will be created the first time it is accessed. Resetting your API Key will invalidate the current key.

Figure 2: After activating the realm, generate an API key to publish modules later.

can then be re-published with the `Publish-Module` command, as before. Nexus automatically detects the update and assigns the new version correctly. You can see the different versions of the module in the Browse menu in the Nexus web interface.

Best Practices for Modules

The technical implementation of a module repository is only the first step. Equally important is establishing organizational rules to ensure long-term productivity. Besides internal modules, you often need access to third-party modules from the PowerShell Gallery. Sonatype Nexus offers the option of using proxy repositories so that external modules are only available after they have been checked from the internal repository.

In practice, a check could include a code security assessment and testing for known vulnerabilities; it should take place in an isolated environment to mitigate potential risk. The modules are only released for use after completing these checks with OK results.

Defining clear processes for deploying and updating internal modules is essential: Who is authorized to publish modules, and what standards must the modules meet? Best practices dictate providing detailed help texts for all of a module's functions. You should also ask yourself the following questions:

- How can approval processes be designed to be as efficient as possible?
- How do you handle rollbacks in the event of faulty modules?
- What automated tests should be performed to ensure the quality of the modules?

It makes sense to implement the processes for publishing and testing modules in the style of DevOps pipelines to make sure they run seamlessly and automatically. Automated tests (e.g., with Pester for PowerShell code tests and mocks) are an important part of ensuring module quality.

Standardization of Repository Usage

If you want to make an internal repository accessible to all PowerShell users, several approaches ensure availability on all systems, such as:

```
if (!(Get-PSRepository &lt;Repo-Name&gt; ↩
      -erroraction SilentlyContinue)){
Register-PSRepository &lt;Repo-Name&gt;
}
```

This command can be executed in different ways on all of your systems, and you can use PowerShell remoting to start this command centrally to ensure that the repository is registered on all relevant systems. Alternatively, the repository could be stored in the base image for the systems, to make it available immediately after installation. In a Windows domain environment, a group policy that executes a start script to register the repository is also a good option.

Conclusions

By grouping functions in modules and making these modules available in a central repository, you can create a useful approach for organizations looking to improve their script-based automation processes. The use of technologies such as NuGet and platforms such as Sonatype Nexus ensures secure and efficient management of PowerShell modules, providing teams with tested and approved code modules and, in turn, improving the security and quality of the scripts. Also, clearly defined processes, such as security checks for external modules and standardized approval workflows, further help to consolidate and secure the use of module repositories within the organization. ■

Info

[1] Sonatype Nexus: [https://www.sonatype.com/products/sonatype-nexus-oss-download]

Checking DNS configurations for errors

Right Connections

Correct name resolution is a fundamental prerequisite for trouble-free communication between servers and between servers and users. We look at practical tools for troubleshooting DNS configurations. By Thomas Joos

A Domain Name System (DNS) name resolution configuration that is not carefully considered or simply incorrect will tend to cause a variety of problems on a network. Unstable server-based services on the network are just one direct consequence of a faulty DNS setup.

If you use Active Directory, the functionality of DNS becomes even more important. Numerous examples illustrate why it is advisable to take a closer look at and optimize the DNS configuration on your network. Hacker attacks on name resolution can never be ruled out – the consequences being server system hijacking, users being lured into traps, or data theft. In the following pages, I look at a number of critical steps toward protecting your DNS.

Best Practices Analyzer

After the release of Windows Server 2025, Microsoft expanded the functionality of the Best Practices Analyzer (BPA) to include automatic server role verification, which is one of the integrated features of Server 2025 that you can use in Server Manager to check server roles across the network. Of course, BPA is also included with previous versions, and it can help you check virtually any server role and display the results centrally. One of its many capabilities is putting DNS through its paces. After selecting the *Manage* | *Add Servers* option, you can add all of the Windows servers on the network to Server Manager. Server Manager then sorts the servers by their roles and automatically creates server groups. The fastest way to start and enable BPA for server roles is to enter the PowerShell command:

```
Get-BPAModel | Invoke-BpaModel
```

If you only want to run a scan for a specific role, such as the DNS server, you first need to retrieve the matching model ID. You can then launch the tests

```
Invoke-BPAmodel ➘
  -modelId Microsoft/Windows/DNSServer
```

and check the results (Figure 1). You could also run a targeted search for a string such as *Error*:

```
Get-BpaResult ➘
  -ModelId Microsoft/Windows/DNSServer | ➘
  Where-Object Severity -eq "Error" | ➘
  Format-List Title
```

The command returns a list of all messages containing the *Error* string. Another useful cmdlet is

```
Get-DnsClientCache
```

which reads the server's current DNS cache and diagnoses potential DNS problems, telling you whether the cache contains outdated or incorrect entries (Figure 2). Incidentally, the `ipconfig /displaydns` command also obtains this information. The fastest way to clear the cache is to type `ipconfig /flushdns`. The equivalent in PowerShell would be running the

```
Clear-DnsClientCache
```

cmdlet, which can be helpful because attempts at name resolution first target the cache. In this context, it is also possible to identify attacks on DNS servers by checking the list for fake DNS entries. One form of spoofing injects DNS entries into the cache.

Photo by MW on Unsplash

Testing Name Resolution with nslookup

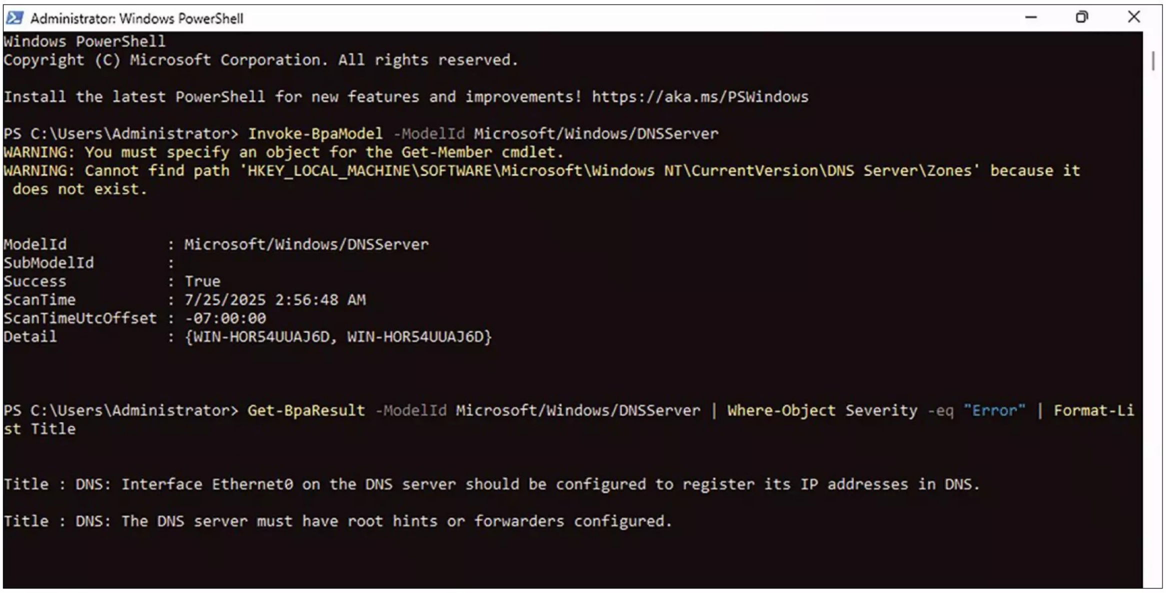
The `nslookup` command-line tool is useful for troubleshooting Microsoft networks or testing Internet access. If it fails to resolve the server name, you can perform a step-by-step error analysis. If you encounter problems in the matching DNS zone, you can register the server's name and IP address by typing

```
ipconfig /registerdns
```

If the error message persists, check whether the server's primary DNS suffix matches the zone name of the DNS zone. Next, determine whether the IP address of the server is correct and whether the entry for the preferred DNS server in the IP settings meets the requirements. Checking the zone properties should reveal whether dynamic updating is permitted. If this is not the case, change the setting to enable updating. You can open the zone properties by right-clicking on the respective zone and selecting *Properties*. If everything is set correctly, the following steps can help you troubleshoot the issue:

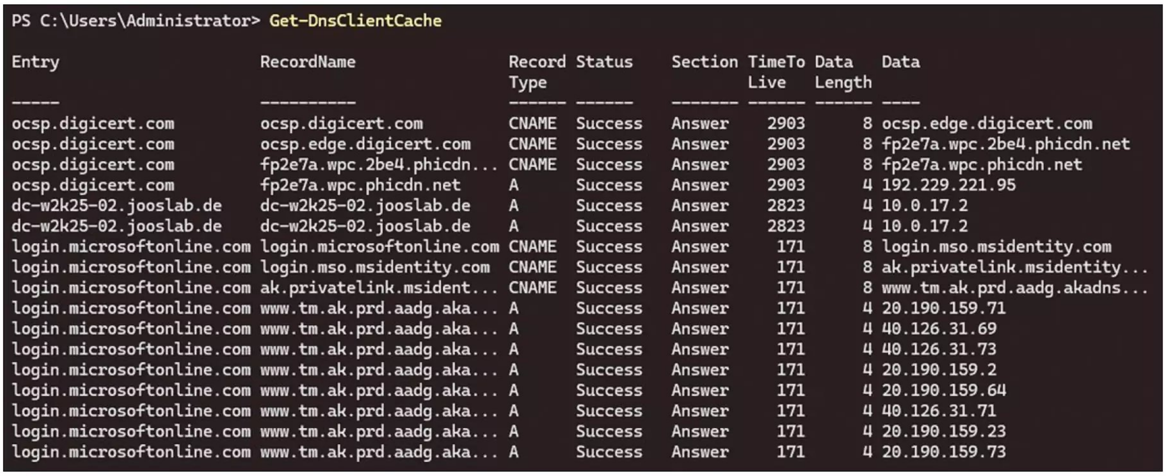
- Check that the correct DNS server is entered as the preferred server in the computer's IP settings.
- Test whether the preferred DNS server manages the zone in which name resolution will take place.
- Determine whether the server manages the zone itself. If this is not the case, check whether a server that can resolve the zone is entered in the server properties.
- Test whether the forwarding server can resolve the zone or whether it also forwards the request.

A server must be present at some point in the forwarding chain that can ultimately resolve the request; otherwise, the client will not be able to



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Administrator> Invoke-BpaModel -ModelId Microsoft/Windows/DNSServer
WARNING: You must specify an object for the Get-Member cmdlet.
WARNING: Cannot find path 'HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\DNS Server\Zones' because it
 does not exist.


ModelId           : Microsoft/Windows/DNSServer
SubModelId        :
Success           : True
ScanTime          : 7/25/2025 2:56:48 AM
ScanTimeUtcOffset : -07:00:00
Detail            : {WIN-HOR54UUAJ6D, WIN-HOR54UUAJ6D}



PS C:\Users\Administrator> Get-BpaResult -ModelId Microsoft/Windows/DNSServer | Where-Object Severity -eq "Error" | Format-Li
st Title


Title : DNS: Interface Ethernet0 on the DNS server should be configured to register its IP addresses in DNS.

Title : DNS: The DNS server must have root hints or forwarders configured.
```

Figure 1: BPA, which comes with Windows Server, lets you identify DNS servers and helps you resolve any issues.



```
PS C:\Users\Administrator> Get-DnsClientCache

Entry                     RecordName                Record Status    Section TimeTo Data   Data
                                                    Type                     Live   Length
-----                     ----------                ------ ------    ------- ------ ------ ----
ocsp.digicert.com         ocsp.digicert.com         CNAME  Success   Answer    2903      8 ocsp.edge.digicert.com
ocsp.digicert.com         ocsp.edge.digicert.com    CNAME  Success   Answer    2903      8 fp2e7a.wpc.2be4.phicdn.net
ocsp.digicert.com         fp2e7a.wpc.2be4.phicdn... CNAME  Success   Answer    2903      8 fp2e7a.wpc.phicdn.net
ocsp.digicert.com         fp2e7a.wpc.phicdn.net     A      Success   Answer    2903      4 192.229.221.95
dc-w2k25-02.jooslab.de    dc-w2k25-02.jooslab.de    A      Success   Answer    2823      4 10.0.17.2
dc-w2k25-02.jooslab.de    dc-w2k25-02.jooslab.de    A      Success   Answer    2823      4 10.0.17.2
login.microsoftonline.com login.microsoftonline.com CNAME  Success   Answer     171      8 login.mso.msidentity.com
login.microsoftonline.com login.mso.msidentity.com  CNAME  Success   Answer     171      8 ak.privatelink.msidentity...
login.microsoftonline.com ak.privatelink.msident... CNAME  Success   Answer     171      8 www.tm.ak.prd.aadg.akadns...
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 20.190.159.71
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 40.126.31.69
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 40.126.31.73
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 20.190.159.2
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 20.190.159.64
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 40.126.31.71
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 20.190.159.23
login.microsoftonline.com www.tm.ak.prd.aadg.aka... A      Success   Answer     171      4 20.190.159.73
```

Figure 2: The DNS cache is an essential tool for identifying problems and cyberattacks.

establish a connection and the name lookup will fail.

Once you have called `nslookup`, you can resolve any server name. If you do not enter a fully qualified domain name (FQDN), but only the computer name, the local computer adds the primary DNS suffix of the computer or the DNS suffixes configured in the IP settings to the name. When `nslookup` is called, a query is also sent to the DNS server stored in the IP settings of the local computer. After resolution, you can query other DNS servers from the local computer:

```
nslookup www.microsoft.de 192.168.178.223
```

In this example, `nslookup` attempts to resolve the host *www.microsoft.de* with 192.168.178.223 as the DNS server. To ensure that `nslookup` displays the DNS server name correctly in Active Directory, the DNS server must be registered in the forward lookup zone of the domain. It also makes sense to create a reverse lookup zone in which the IP addresses of the Active Directory domain are registered.

If you want to query several hosts in succession, replace

```
nslookup <Host> <DNS server>
```

with

```
nslookup -<DNS server>
```

The `<DNS server>` entry stands for the name or IP address of the DNS server that you want `nslookup` to use. If you enter the `nslookup` command without any further parameters, it is possible to change the default server to a specific DNS domain with

```
server 192.168.178.223
```

for the current window in the interface, which means you can use different servers for name resolution and narrow down the root cause of the issue. The next step is to correct the errors.

Understanding DNS Settings in Windows

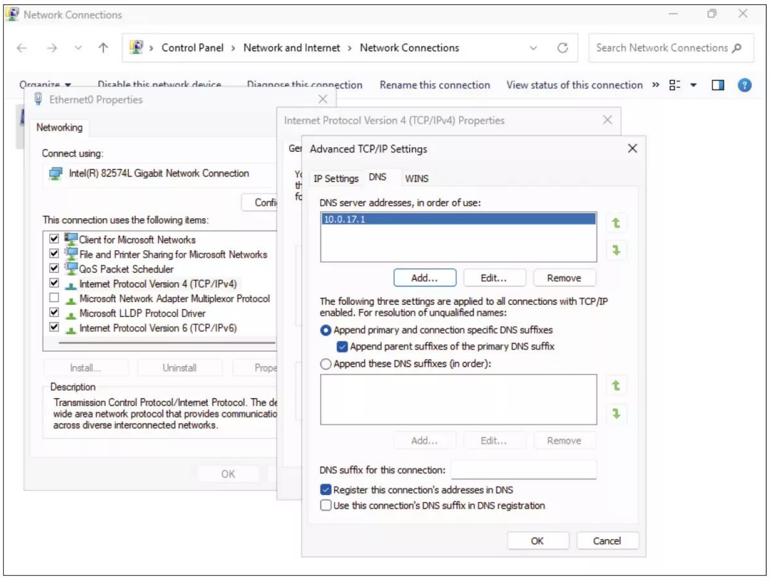
The DNS settings are configured on the *DNS* tab in the properties of a network adapter. The advanced properties of *Internet Protocol Version 4 (TCP/IPv4)* provide various settings, such as integrating Windows into a Windows domain. The settings are prone to incorrect configurations, so you should check them after setup or if you encounter problems with name resolution. The following options are particularly important for name resolution in a DNS infrastructure:

- *Append primary and connection specific DNS suffixes*: When resolving computer names, the computer automatically attempts to append the configured primary DNS suffix of its own computer name. For example, if you want to resolve computer name *dc01*, Windows will attempt to resolve the name to *dc01.contoso.int* if the primary DNS suffix of the computer is *contoso.int*.
- *Append parent suffixes of the primary DNS suffix*: The names of parent domains are also used in name resolution. For example, if you want to resolve server name *dc05* in a subordinate domain named *muenchen.de.contoso.int*, the computer first attempts to resolve it with *dc05.muenchen.de.contoso.int* if this is the primary DNS suffix of the computer. It then attempts to resolve the name with *dc05.de.contoso.int* and then *dc05.contoso.int*, because these domains are parent domains of the *muenchen.de.contoso.int* domain.
- *DNS suffix for this connection*: You also have the option of entering another DNS suffix of your choice. If the computer cannot find the name entered on its configured DNS server from its own primary DNS suffix, it tries the DNS suffix in this field. For example, if you want to resolve server *dc06*, Windows first tries to resolve it to *dc06.contoso.int*, given that this is its primary DNS suffix. If you enter a suffix in this field in the form *muenchen.de.microsoft.int*, the server will also attempt to resolve the name to *dc06.muenchen.de.microsoft.int*.
- *Register this connection's addresses in DNS*: This option is enabled by default. A DNS server has the option of registering entries dynamically, which eliminates the need to create host entries manually. As soon as a computer logs on to the network, it attempts to register its FQDN automatically with the configured DNS server, unless this option is disabled. This point is very important for internal name resolution in an AD network.

In addition to the options enabled by default, this window contains additional configuration options that you can enable as needed:

- *Append these DNS suffixes (in order)*: This option is often used to resolve names in multisite environments. To do so, enter all the sites in the multisite environment in the correct order to support correct name resolution in Active Directory. This option lets you configure DNS suffixes that are used to resolve incomplete computer names. Neither the primary DNS suffix of the server nor the DNS suffixes of this connection are used. The DNS suffixes are appended in the order you configure in this field. Make sure that the DNS suffix of the Windows domain in which this server is a member is the first in the list.
- *Use this connection's DNS suffix in DNS registration*: This option registers the server name in the DNS with its computer name and primary DNS suffix (i.e., its FQDN). Moreover, the name is also registered on the DNS server with the DNS suffix configured in the DNS suffix area for this connection. This option is not enabled by default.

If you want to resolve server names in different DNS zones quickly and efficiently, enable the *Append*



Connect using:
Intel(R) 82574L Gigabit Network Connection

This connection uses the following items:
☑ Client for Microsoft Networks
☑ File and Printer Sharing for Microsoft Networks
☑ QoS Packet Scheduler
☑ Internet Protocol Version 4 (TCP/IPv4)
☐ Microsoft Network Adapter Multiplexor Protocol
☑ Microsoft LLDP Protocol Driver
☑ Internet Protocol Version 6 (TCP/IPv6)

Description
Transmission Control Protocol/Internet Protocol. The de
wide area network protocol that provides communicatio
across diverse interconnected networks.

DNS server addresses, in order of use:
10.0.17.1

The following three settings are applied to all connections with TCP/IP enabled. For resolution of unqualified names:
● Append primary and connection specific DNS suffixes
☑ Append parent suffixes of the primary DNS suffix
○ Append these DNS suffixes (in order):

DNS suffix for this connection:
☑ Register this connection's addresses in DNS
☐ Use this connection's DNS suffix in DNS registration

Figure 3: DNS configuration is prone to errors and offers many setting options.

these DNS suffixes (in order) option on the server in the Advanced TCP/IP Settings dialog (**Figure 3**). Next, enter the namespace of your own tree first and then append the namespaces of the other trees. The purpose of this configuration is to resolve servers quickly in the other trees.

For example, to resolve domain controller *dc1* in the *contoso.int* tree, you need to enter *dc1.contoso.int* if your server is not a member of this tree. This setting improves the stability of name resolution in AD environments. Many administrators enter a new static host entry on their DNS server that points to the IP address of the server in the other namespace. However, this procedure is incorrect, even though it sort of works. In this case, the correct DNS name of the corresponding server is not returned; rather, the server name with the zone of the DNS server in which the server was entered as a host is returned.

Checking SRV Records

In addition to its host A name (e.g., *dc01.contoso.int*), each domain controller in Active Directory is assigned a matching CNAME that stands for the Directory System Agent (DSA) object of its NTDS settings. The DSA object is listed as a service (SRV) record in DNS below the *_msdcs* node of the domain zone. The CNAME is equivalent to the GUID of the DSA object. Domain controllers do not attempt to resolve their replication partners with the conventional host A entry but use the stored CNAME instead. If replication does not work because entries are missing below the AD DNS domain, you can restore them by entering the command

```
dcdiag /fix
```

at the command prompt. After doing so, it makes sense to check the CNAME registered for the server. The Linux `dig` tool lets you run specific DNS queries. It is available in Kali Linux, for example. An example of a query for an SRV record and its output could look like:

```
dig _ldap._tcp.jooslab.de SRV
_ldap._tcp.jooslab.de. 3600 ➘
  IN SRV 0 100 389 dc1.jooslab.de
_ldap._tcp.jooslab.de. 3600 ➘
  IN SRV 0 100 389 dc2.jooslab.de
```

The result shows that the two domain controllers, *dc1* and *dc2*, are correctly registered for LDAP queries. In addition to Active Directory, `dig` can also be used in other network environments to check DNS entries such as MX records for email or A and AAAA entries for hostnames. Simple queries such as

```
dig jooslab.de MX
```

help analyze mail servers and identify possible misconfigurations.

DNS Analysis with dnsrecon

The Python-based `dnsrecon` **[1]** tool has DNS analysis features and is useful for managing and securing AD-based networks. The `dnsrecon` tool helps you discover potential vulnerabilities in the DNS system and obtain detailed information about DNS entries. The tool supports a variety of scans, including standard queries, brute force attacks on subdomains, and searches for SRV records, which are important in Active Directory environments. For example, the command

```
dnsrecon ➘
  -d jooslab.de ➘
  -D /usr/share/wordlists/dnsmap.txt ➘
  -t std ➘
  --xml dnsrecon.xml
```

first runs a standard lookup that lists the SOA, NS, A, AAAA, MX, and SRV entries. At the same time, a brute force search for subdomains is performed with the use of a previously defined word list. The results are then stored in an XML file. In an Active Directory environment, this approach allows essential SRV records such as *_ldap._tcp* or *_kerberos._tcp* to be identified. It is also possible to check zone transfers to ensure that no unauthorized DNS servers can access sensitive information. The integration of `dnsrecon` into security checks helps you identify and eliminate risks arising from incorrectly configured DNSSEC settings or the absence of recursion restrictions.

Securing DNS with fierce

The `fierce` tool **[2]** is suitable for passive and active network analysis, especially in Active Directory environments. This tool is also available in Kali Linux. It collects information about DNS and network structures with the aim of revealing potential attack vectors. Identifying subdomains, hostnames, and IP ranges helps you detect insecure or misconfigured DNS entries, for example with:

```
fierce -dns jooslab.de
```

As part of the check, `fierce` analyzes the *jooslab.de* domain for subdomains and other hosts associated with the domain. The tool uses brute force methods and reverse lookups to determine IP addresses and their assignments. On AD networks, this method can be helpful for detecting previously undiscovered services such as unauthorized domains and their domain controllers or undocumented systems. Fierce also checks the DNS servers for potential misconfigurations, for example, with regard to missing restrictions for zone transfers.
The tool also lets you use a specific DNS server for the query:

```
fierce -dns jooslab.de ↩
       -dns-servers ns1.jooslab.de
```

To determine the identity of unknown hosts or systems on a network, `fierce` can work with an IP range:

```
fierce --range 192.168.1.1-192.168.1.254
```

The tool combines hostname and DNS scans to generate a comprehensive overview. The command

```
fierce ↩
  -dns jooslab.de ↩
  -file /usr/share/wordlists/dnsmap.txt
```

lets you detect hidden subdomains and their associated IP addresses.

Analyzing DNS Server Hierarchies with dnstracer

The `dnstracer` command can reveal the DNS hierarchies of a network by tracing the path of a DNS query through the servers involved. In Active Directory, environments this function is useful for ensuring that authoritative and forwarding DNS servers are working correctly:

```
dnstracer -s ns1.jooslab.de jooslab.de
```

The output shows how the query is resolved from the root zone through the top-level domain (TLD) servers to the authoritative DNS server. The analysis reveals problems, such as incorrectly configured forwarding or unreachable servers. The tool can also be used in non-Active Directory environments to visualize complex DNS structures (e.g., when configuring load balancers or failover-capable DNS servers).

Manipulating DNS Responses with dnschef

The `dnschef` [3] tool sets up bogus DNS servers and is used on Active Directory networks to test the effects of DNS manipulation on applications and services. The following example shows how to start a DNS server that redirects all requests to the *jooslab.de* domain to a specific IP address,

```
dnschef ↩
  --fakeip 192.168.1.100 ↩
  --fakedomain jooslab.de
```

which allows you to simulate scenarios such as DNS spoofing to evaluate the effectiveness of security measures. In other networks, `dnschef` is often used in test environments to analyze the effects of DNS errors on users or services.

Conclusions

DNS is an essential component of virtually all networks, especially in Active Directory environments. With the built-in tools and optional add-ons it provides, vulnerabilities can be identified and resolved in a targeted way. Even if the network appears to be stable at first glance, it is still advisable to check the configurations. The tools and procedures presented in this article can be helpful in this regard. ■

Info

[1] dnsrecon: [https://github.com/darkoperator/dnsrecon]
[2] fierce: [https://www.kali.org/tools/fierce/]
[3] dnschef: [https://www.kali.org/tools/dnschef/]

The Author

Thomas Joos is a freelance IT consultant and has been working in IT for more than 20 years. In addition, he writes hands-on books and papers on Windows and other Microsoft topics. Online you can meet him on [http://thomasjoos.spaces.live.com].

■

Linux Magazine Subscription

Print and digital options
12 issues per year

Lomiri: Rising from the ashes of Ubuntu's Unity desktop
LINUX MAGAZINE
Pen Testing
Find the gaps before intruder finds them
Gimp 3.0: The great free image editor rolls out a major update
Cleaning Up
Find the forgotten files that are clogging your system
ChimeraOS: Play Steam, GoG, and Epic games with this versatile gaming OS
Inform: Interactive fiction in Linux
10 TANTALIZING FREE APPS!
Beyond WiFi: Build a long-range sensor network that reaches the back of your garden
Indie Game Studio
Equip your shop with all open source tools
Warp: This cool terminal app adds natural language and other AI features
Wazuh: Watch logs and check for vulnerabilities
Snapshot: Write a script that finds photos using geodata
Tricks with VirtualBox: Build a virtual playground to experiment with custom GRUB 2 settings
Just: Easy automation with this handy command runner
10 TANTALIZING FOSS FINDS!
WWW.LINUX-MAGAZINE.COM

SUBSCRIBE
shop.linuxnewmedia.com/shop

Expand your Linux skills:

- In-depth articles on trending topics, including Bitcoin, ransomware, cloud computing, and more!
- How-tos and tutorials on useful tools that will save you time and protect your data
- Troubleshooting and optimization tips
- Insightful news on crucial developments in the world of open source
- Cool projects for Raspberry Pi, Arduino, and other maker-board systems

Go farther and do more with Linux, subscribe today and never miss another issue!

Follow us

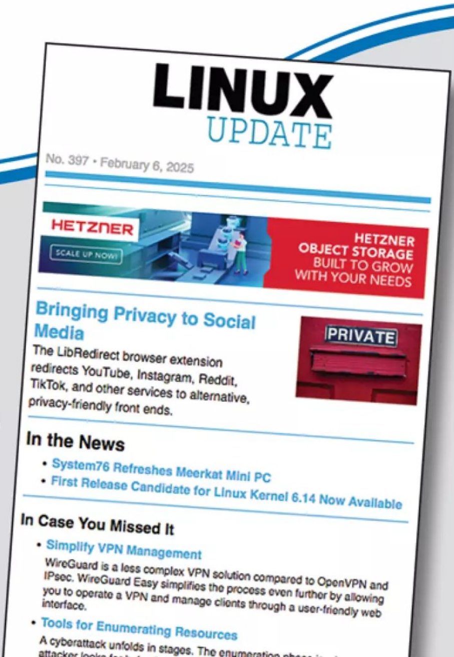
@linux-magazine.com
Linux Magazine
@linuxpromagazine
@linuxmagazine@fosstodon.org

Need more Linux?

Subscribe free to Linux Update

Our free Linux Update newsletter delivers insightful articles and tech tips to your inbox every week.

bit.ly/Linux-Update



LINUX UPDATE

No. 397 • February 6, 2025

HETZNER
SCALE UP NOW!
HETZNER OBJECT STORAGE BUILT TO GROW WITH YOUR NEEDS
PRIVATE

Bringing Privacy to Social Media

The LibRedirect browser extension redirects YouTube, Instagram, Reddit, TikTok, and other services to alternative, privacy-friendly front ends.

In the News

- System76 Refreshes Meerkat Mini PC
- First Release Candidate for Linux Kernel 6.14 Now Available

In Case You Missed It

- **Simplify VPN Management**
  WireGuard is a less complex VPN solution compared to OpenVPN and IPsec. WireGuard Easy simplifies the process even further by allowing you to operate a VPN and manage clients through a user-friendly web interface.
- **Tools for Enumerating Resources**
  A cyberattack unfolds in stages. The enumeration phase is

Photo by Amol Tyagi on Unsplash_Key.jpg

Securing email communication with GnuPG

Key Master

We look at certificate management with a Web Key Directory and a directory service and investigate automated authentication with GnuPG actium. By Heike Jurzik

GNU Privacy Guard (GnuPG) [1] is an open source tool used worldwide for encrypting and signing mail and files. Its implementation and use can be challenging when it comes to distributing certificates on a large scale and authenticating their origin. Flexible trust models such as the hierarchical model with a trusted key as the trust anchor and optional trusted introducers for domain-specific authentication simplify management. The trusted key (certificate authority (CA) certificate) authenticates keys throughout the organization, while trusted introducers (intermediate certificates) add trust relationships to specific domains in a targeted way. Further details and background information can be found in the GnuPG white paper on certificate management [2].

GnuPG forms the basis for other tools designed for different security requirements: GnuPG Desktop [3] and Gpg4win [4] are user-friendly desktop versions. GnuPG VS-Desktop [5] is available for particularly secure use in government agencies and companies in which security and data integrity are paramount, especially when it comes to information with EU-RESTRICTED and NATO-RESTRICTED information [6].

Provisioning Keys in the WKD

A Web Key Directory (WKD) provides a simple and secure way to make public keys for email addresses available over HTTPS. The underlying standard was first implemented in GnuPG version 2.1.12 and has been enabled by default since version 2.1.23. The Web Key Service (WKS) complements this process by automating the management and publication of keys in a WKD, but it is not mandatory. Many organizations and individuals simply set up a WKD. The advantage is that you can ensure that the public keys of the email users in your domain are always accessible over HTTPS.

This method is not only suitable for smaller organizations or individuals, but also for larger companies with several thousand employees. To set up a WKD, all you need is a web server for your domain (e.g., Apache or NGINX). The server must have a valid SSL/TLS certificate; you then create a static directory there that is updated regularly, which can easily be done in parallel with content management systems, because the WKD is located in a dedicated directory.

Configuring the Web Server

To put a WKD into operation, you first need to create the required folder structure on the web server, where the public keys will be located later:

```
mkdir -p /var/www/virtual/example.org/ ↩
 .well-known/openpgpkey/hu
```

WKD uses hashes instead of plain text names to avoid problems with file name encoding. To let WKD clients see that your server offers a Web Key Directory, you need to create an empty file as a hint for the clients:

```
touch /var/www/virtual/example.org/ ➘
  .well-known/openpgpkey/policy
```

Make sure you change the path to the `DocumentRoot` directory to suit your environment. Next, configure your web server to define the `.well-known/openpgpkey` directory as the storage location for PGP keys. You can use the configuration shown in **Listing 1** for Apache. This arrangement ensures that the PGP keys are delivered as binary data (`application/octet-stream`) and that access is allowed from any domain (CORS, cross-origin resource sharing).
The configuration looks very similar for NGINX:

```
## WEB KEY DIRECTORY ##
location /.well-known/openpgpkey/ {
  types { }
  default_type application/octet-stream;
  add_header Access-Control-Allow-Origin ➘
    "*" always;
}
```

The instructions specify that access to the files is possible from any domain.

Determining the WKD Hash

To generate the hash that WKD uses for the key files, run the command

```
gpg --with-wkd-hash ➘
    --fingerprint name@example.org
```

which produces output similar to

```
ed25519 2022-08-29 [SC] ➘
  [expires: 2027-08-28]
uid [ultimate] Name name@example.org
updgef8corpy81hia1rhd8npqiti6nzf ➘
  @example.org
```

In the output you will see a 32-character string to the left of the domain part of the email address under each UID line with an email address. This string is the WKD hash – in this example, *updgef8corpy81hia1rhd8npqiti6nzf*. Now export the public key in binary format (without the ASCII armor) and name the file after the WKD hash:

```
gpg --no-armor --export name@example.org > ➘
  updgef8corpy81hia1rhd8npqiti6nzf
```

Next, save the exported file in the `.well-known/openpgpkey/hu` directory on your web server. Once the public key is in the right directory, your WKD is ready to use. To export multiple certificates for a specific domain, use the command

```
gpg --list-options show-only-fpr-mbox ➘
     -k @example.com | gpg-wks-client ➘
    --install-key
```

Assign responsibility for maintaining the WKD to a trusted person in your organization. Employees can then simply send their certificates to the maintainer, who adds them to the WKD. To do this, the maintainer imports the keys locally into their keychain and then uploads them automatically to the web server with a simple command or script.
To mirror the certificates regularly (e.g., once or twice a day) a script with a cron job or a scheduled task is recommended. The script first needs to empty the local mirror directory with the `rsync --delete` option, for example, so that revoked keys automatically disappear from the WKD.

Internal Key Server

A directory service such as Active Directory Lightweight Directory Services (AD LDS) on Windows or OpenLDAP on Linux is ideal for providing certificates with the Lightweight Directory Access Protocol (LDAP) in your organization. This setup is useful if you have 25 or more users and is practically indispensable for more than 50 users; importing certificates manually can quickly become too time-consuming.
To provide certificates from a directory service, you need at least GnuPG 2.2.28 and two special LDAP schema files in LDAP Data Interchange Format (LDIF). The GnuPG developers provide configurations for Windows **[7] [8]** and Linux **[9] [10]** that are based on the PGP schema developed in the late 1990s. The developers have extended the schema definitions to include important attributes that make it possible to search for certificates by their fingerprints and to store information about subkeys and UTF-8-encoded email addresses.

AD LDS Setup on Windows

To import the schema files into the LDAP directory service, download them from the manufacturer's website. On Windows, pop up a command prompt as administrator and use the `ldifde` tool to import them:

```
ldifde -i ➘
       -f C:\<path to file>gnupg-ldap-➘
            ad-schema-v1.ldif ➘
       -s <server name> ➘
       -t <port number> ➘
       -c "DC=EXAMPLEDC" ➘
          "#configurationNamingContext"
ldifde ➘
  -i ➘
  -f C:\<path to file>gnupg-ldap-ad-➘
       init-v1.ldif ➘
  -s <server name> ➘
  -t <port number> ➘
  -c "DC=EXAMPLEDC" ➘
     "CN=keyserver,DC=demo,DC=orga,DC=de"
```

Replace the placeholder with the name of your AD server (e.g., `localhost`). The port number after `-t` is only required if you are using a port other than the default port (which

Listing 1: Apache Configuration for WKD

```
<Directory "/var/www/virtual/.well-known/openpgpkey">
   Options FollowSymLinks
   AllowOverride None
   # Set the default content type to binary/octet-stream
   AddType application/octet-stream .*
   # CORS header to allow access from all origins
   <IfModule mod_headers.c>
      Header always set Access-Control-Allow-Origin "*"
   </IfModule>
</Directory>
```

is 389 for LDAP or 636 for LDAP over SSL). The `-c` parameter replaces placeholders within the LDIF files at runtime; for example, `#configurationNamingContext` is replaced with `DC=EXAMPLEDC`. In the second command, `#configurationNamingContext` is replaced by the full distinguished name (DN) of the key server (`CN=keyserver,DC=demo,DC=orga,DC=de`) to integrate the entries into the specific AD structure.

Finally, you will need to define permissions such that domain users on the network can access the key server. To do this, use the `ldp.exe` tool to manage access rights and set up full read access and extended write permissions. For more details, refer to the LDS installation guide **[11]**.

LDAP Setup on Linux

On Linux, you need to install the schema as the admin user (root) on your LDAP server with the command

```
ldapadd -v -Y EXTERNAL -H ldapi:/// ⮐
  -f ./gnupg-ldap-schema.ldif
```

The `-v` ensures verbose output, and `-Y EXTERNAL` defines the authentication mechanism (local socket authentication `ldapi` in this example). The `-H ldapi:///` refers to the standard OpenLDAP socket (typically `/var/run/slapd/ldapi`). The schema is defined by the `-f` option.

The next step is to link the new schema to your directory tree to create entries for storing the certificates and metadata so that GnuPG can find the data. To do so, edit the downloaded `gnupg-ldap-init.ldif` file and replace all relative DNs (RDNs) with `dc` with your own domain.

Repeat this process for all other `dc` entries in the file. If you are using a three-level domain, add another `dc` entry as appropriate, just as you do when setting up OpenLDAP; then, run the modified file with the command

```
ldapadd -v -x -H ldapi:/// ⮐
  -D 'cn=admin,dc=example,dc=com' ⮐
  -W -f ./gnupg-ldap-init.ldif
```

Be sure to adapt the RDN attributes to your configuration here, too. The `-x` option gives you simple authentication, `-D` specifies the bind DN for connecting to the LDAP directory, and `-W` prompts for the administrator password.

Path to Directory Service

Various approaches can be employed for connecting Windows computers to the directory service. One option is to create an `NTDSKeyserver` entry below `Computer\HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\GNU\GnuPG` in the registry and store the server address there. This option only works for the GnuPG VS-Desktop variant. If needed, you can distribute the registry entry throughout the organization by group policies. Alternatively, just modify the settings of the Kleopatra certificate manager program for the users and enter the server address there.

Linux users can also store the directory service in the Kleopatra graphical certificate manager. Alternatively, you can edit the `dirmngr.conf` configuration file, which is typically located in the GnuPG home directory, `~/.gnupg`. Add a line like

```
keyserver ldap://ldap.example.com
```

Centralized management by LDAP is less time-consuming than sending certificates manually and more practical than WKD, because everything is provisioned in a single place.

As an admin, you will want to define a clear process for uploading certificates to the LDAP server and decide up front which users should have read-only or write access.

If you have already set up a WKD in your organization, you can use the `gpg-wks-client` tool to mirror certificates stored in a directory service automatically to the WKD:

```
gpg-wks-client -mirror example.com
```

This command searches the directory service for the certificates and then creates matching entries for the users in the WKD. By doing this step, the certificates are available both in LDAP and also directly in the Web Key Directory over HTTPS.

Certificate Distribution Authentication

Thus far, I have focused on how you can distribute certificates within your organization. However, this alone is often not enough – especially if you want to ensure that the keys come from the right people. Authentication now comes into play.

In smaller teams or with a limited number of users, it might be fine for one maintainer in the organization to check and authenticate the keys manually. This person could, for example, publish the key with the WKD and at the same time confirm the identity of the person whose key they are managing. Trust plays a central role here: You need to be able to rely on this person to check the fingerprints conscientiously and handle authentication correctly.

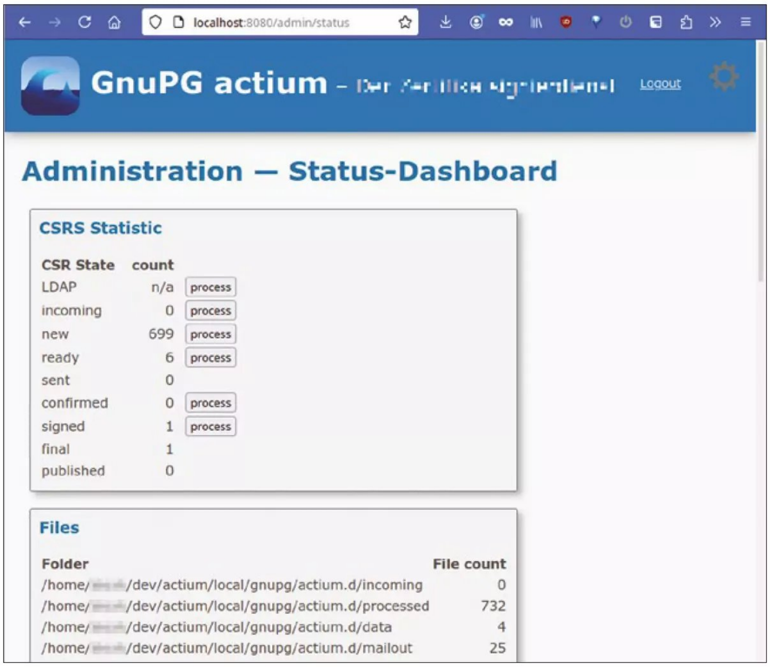
The best way is to use your own trusted key for certification in your organization. Administrators can enter the fingerprint of this key directly in Active Directory. Alternatively, you can store the fingerprint for GnuPG or Gpg4win in the `gpg.conf` file (the `trusted-key <CA-FINGERPRINT>` option), for example, in `C:\%APPDATA%\gnupg`.

In larger organizations or more complex environments, manual authentication quickly becomes impractical, which is where an automated solution comes into play. Enter GnuPG actium.

Automation with GnuPG actium

The GnuPG actium **[12]** program, written in Go, simplifies certificate management in your organization. It automates the authentication process with an LDAP directory service and processes Pretty Good Privacy (PGP) certificates that are available as individual files. Licensed under GPLv3, the tool runs on Windows and Linux. The current version, 1.2.1, was released in August 2025. Actium (including the man page) is available from the manufacturer (**info@gnupg.com**).

GnuPG actium (Figure 1) monitors the LDAP directory service; detects new certificates, retrieves them automatically, and validates them; and sends an encrypted confirmation email to users. To activate the confirmation link you need access to the secret key. After confirmation, actium certifies the certificate with the organization's trusted key and republishes it in the directory service. The tool checks user IDs and their assignments to the domain or directory service structure to ensure that the certificates originate from authorized users. Installation and setup can be done at the command prompt or command line; a web interface in the browser is available for admin tasks.



Administration — Status-Dashboard

CSRS Statistic

CSR State	count	
LDAP	n/a	process
incoming	0	process
new	699	process
ready	6	process
sent	0	
confirmed	0	process
signed	1	process
final	1	
published	0	

Files

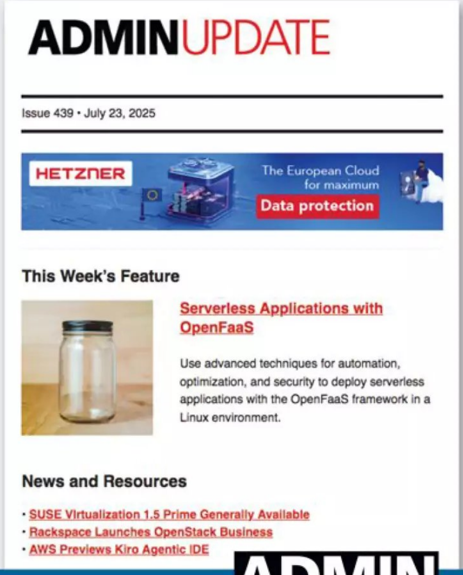
Folder	File count
/home/[redacted]/dev/actium/local/gnupg/actium.d/incoming	0
/home/[redacted]/dev/actium/local/gnupg/actium.d/processed	732
/home/[redacted]/dev/actium/local/gnupg/actium.d/data	4
/home/[redacted]/dev/actium/local/gnupg/actium.d/mailout	25

Figure 1: The GnuPG actium status dashboard gives you a quick overview of the certificate processes in progress.

Installing Actium on Windows and Linux

To install actium on Windows, first make sure you have installed GnuPG

ADMIN UPDATE



TOO SWAMPED TO SURF?

ADMINUPDATE

Issue 439 • July 23, 2025

HETZNER The European Cloud for maximum Data protection

This Week's Feature

Serverless Applications with OpenFaaS

Use advanced techniques for automation, optimization, and security to deploy serverless applications with the OpenFaaS framework in a Linux environment.

News and Resources

- SUSE Virtualization 1.5 Prime Generally Available
- Rackspace Launches OpenStack Business
- AWS Previews Kiro Agentic IDE

Our ADMIN Update newsletter delivers news and technical articles to your inbox every week.

Subscribe free, and you'll discover:

- practical articles on security, cloud computing, DevOps, and more
- IT industry news
- job listings for IT professionals
- IT industry events
- and more!

bit.ly/ADMIN-Update

ADMIN Network & Security

VS-Desktop (version 2.4.3) or Gpg4win (version 4.4.1 or newer). The recommended approach is to create a separate user account for actium. If you plan to check the user IDs of the keys against your Active Directory, this account needs to be a member of the matching Windows domain.

Copy the `actium.exe` and `gpgme-w32spawn.exe` files to a suitable folder (e.g., `C:\Program Files (X86)\gnupg-actium`). Actium uses the GnuPG keyring; the recommended approach is to set up a separate GnuPG folder to avoid any effect on your personal GnuPG configuration. A good choice of location is `C:\gnupg-actium`. Create the `actium.d` folder and, optionally, `actium.d\certs` subfolder for TLS certificates.

To ensure the tool has access to its files, add the path to `actium.exe` to the `PATH` environment variable and create a `GNUPGHOME` variable that points to the `gnupg-actium` folder. Alternatively, you can create a batch file (e.g., `start-actium.bat`) in the main directory to launch actium with the required environment variables. Now type

```
actium initialize
```

to create the required directories and files in the `actium.d` folder.

To start actium automatically as a service, you can use the Task Scheduler. Because the software is fairly frugal in terms of resource consumption, it is a good idea to run the program on the same server that runs the LDAP directory service.

On Linux, you also need GnuPG, including the `dirmngr` program and *GPGME* library. Define a dedicated GnuPG home directory (e.g., `/home/actium/gnupg`), create an `actium.d` subdirectory inside it, so you don't interfere with your regular GnuPG home. This directory acts as the home folder and must exist; otherwise, actium will not start for security reasons. Now initialize the software with the `--initialize` parameter to create the folder structure and the required files.

Listing 2: actium Configuration

```
certification-key <CA-Fingerprint>
certification-domains <Domain>...
domain-validation valid
# check against Active Directory
# (only on Windows, when actium user is logged in to AD)
ad-validation all
# publish back to LDS
publish-processor ldap
# mail delivery
mail-delivery smtp
mail-smtp-server <mail server>
```

GnuPG actium works with common init systems and can therefore easily be launched automatically at system startup, monitored, and restarted when needed to ensure continuous availability.

actium Configuration and Management

Next, create the `actium.conf` file, where you enter the LDAP and SMTP servers and the trusted key required for authentication. Listing 2 shows a minimal configuration. Enter this information about the mail server and directory service for actium. Details of all options can be found in the actium man page. The LDAP server is configured as described before.

After the initial installation and after making changes to the configuration, it is a good idea to restart actium at the command line. Check the output for issues and error messages. The software only writes its log (defined in the `actium.conf` file) to the logfile after successfully parsing the configuration.

The tool's web interface offers clear-cut, browser-based control and provides an overview of the entire certificate process. The interface lists running processes and the status of new and previously signed certificates. On the status dashboard, you can see at a glance how many certificates are in the inbox, which ones have already been checked and are ready, and which ones have been prepared for sending. If needed, you can intervene in each process from the web interface. You can pause, recheck, or send certificates directly. One particularly practical feature is that all the commands you would otherwise execute on the command line can be entered and controlled here.

Conclusions

Whether you use WKD, LDAP, or GnuPG actium, the right choice of key and certificate management adds security and efficiency to your everyday work. Each of the services shown here has its strengths, from uncomplicated key retrieval to fully automated certificate authentication. The important thing is to choose tools that fit your organizational structure and security requirements to best secure your digital communications. ■

Info

[1] GnuPG: [https://gnupg.org]

[2] White paper on certificate management: [https://gnupg.com/doc/whitepaper_certificate-management_gnupg_en.pdf]

[3] GnuPG Desktop: [https://gnupg.com/gnupg-desktop.html]

[4] Gpg4win: [https://www.gpg4win.de/index.html]

[5] GnuPG VS-Desktop: [https://www.gnupg.de/gnupg-vs-desktop.html]

[6] Security with VS-NfD Certification: [https://gnupg.com/gnupg-vs-desktop.html]

[7] LDIF file (schema) for Windows: [https://gnupg.org/misc/gnupg-ldap-ad-schema-v1.ldif]

[8] LDIF file (init) for Windows: [https://gnupg.org/misc/gnupg-ldap-ad-init-v1.ldif]

[9] LDIF file (schema) for Linux: [https://gnupg.org/misc/gnupg-ldap-schema.ldif]

[10] LDIF file (init) for Linux: [https://gnupg.org/misc/gnupg-ldap-init.ldif]

[11] LDS installation guide: [https://gnupg.com/doc/install_lds_gnupgvsd_en.pdf]

[12] GnuPG actium: [https://www.gnupg.de/gnupg-actium.html]

The Author

Heike Jurzik has been writing about the free operating system Linux and other Open Source Topics for more than 20 years. Her first Linux command line articles were published in the late 1990s – when Linux itself was still young. Since then, she has written several books, technical articles, blog posts and manuals in German and English.

CLEAR OFF YOUR BOOKSHELF WITH DIGITAL ARCHIVES

Complete your collection of *Linux Magazine* and *ADMIN Network & Security* with our Digital Archive Bundles.

You get a full year of issues in PDF format to access at any time from any device.

2024 Archives Available Now!

shop.linuxnewmedia.com/shop

Image © rukanoga, 123RF.com

Building a custom Cyberdeck

Hit the Deck

Investigate data center performance issues in style!

By Federico Lucifredi

The idea of a dedicated terminal to investigate configuration inconsistencies and performance issues in a data center not thoughtfully configured with rackmount monitors and keyboards is an old one for me – I have been trying such tricks since the Acer Aspire One [1] made its first appearance.

A low-cost netbook seemed like the ideal, all-purpose monitoring instrument back when its product line launched nary 20 years ago (Figure 1). Its underpowered early Atom CPU cannot support a modern graphical environment, but the intended use as a terminal in `init 3` mode moots that concern entirely. Sadly, I am unable to fully enjoy my sharply configured, cutting-edge, Rolling Rhino-powered device [2] because my large hands are a poor match to a netbook's smaller keyboard – especially when typing fast and furious as rapid troubleshooting demands. It's time to give this project another try, but with the modern hardware of the 2020s!

The Keyboard Is the Computer

The Raspberry Pi team has been experimenting with the keyboard-integrated computer format so popular back in the age of Commodore's Amiga and C64 machines, launching the Pi 400 [3] and its recent upgrade, the Pi 500 [4], soon after the launch of the Raspberry Pi 5 itself (Figure 2). These machines feature the hardware of the corresponding single-board computer (SBC), laid out in a custom board designed for better thermal behavior in the closed environment found inside a keyboard's case.

Although the Pi 400 is essentially a Raspberry Pi 4, the Pi 500 differs from its parent by lacking a PCIe interface. The ports exposed at the back are essentially the same laid out in different arrangements (Figure 3), but the Pi 500 increased RAM to 8GB and CPU speed to 2.4GHz – from 4GB and 1.8GHz, respectively, in the Pi 400. The core design was also upgraded from the ARM Cortex-A72 to Cortex-A76.

Going Cyberpunk

In addition to the Pi 500 itself and its official Raspberry PI-certified SD card [7], I am partial to The Pi Hut's sleeve expressly designed for these keyboard computers. Although the felt fashion may be a little 2020 hipster, that is where it stops, because the next part for the build comes straight from Adafruit zaibatsu and could well be sourced from the Deliverator's own arachnofiber backpack (if unfamiliar with the cyberpunk terminology coloring this article, please refer to the minimal dictionary found in Table 1). The Cyberdeck HAT (Figure 4, Table 2) carries unusual, angled GPIO connectors to support and power a display plugged directly into the back of the keyboard, as well as two Stemma and Stemma QT plugs to add additional wire sensors and controls [9] [10]. I selected

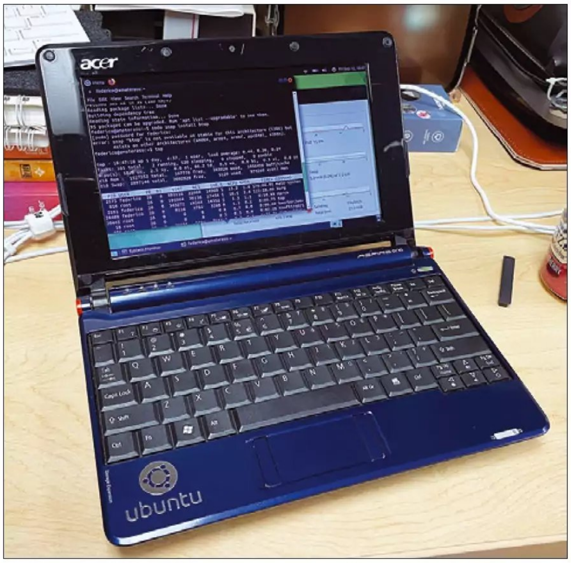


Figure 1: The diminutive (and shiny!) Acer Aspire One was a first attempt at a compact terminal. The keys' minute size was an ill fit to my full-size hands.

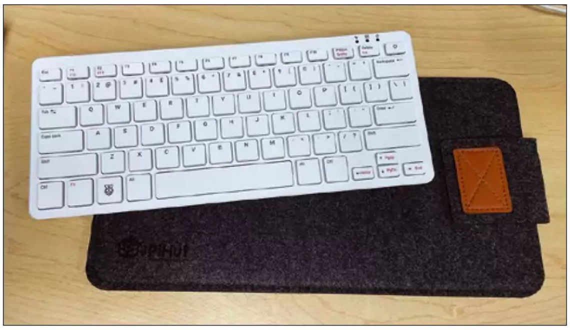


Figure 2: This Pi 500 keyboard computer is about to become a slicing deck.

Lead Image © Lucy Baldwin, 123RF.com

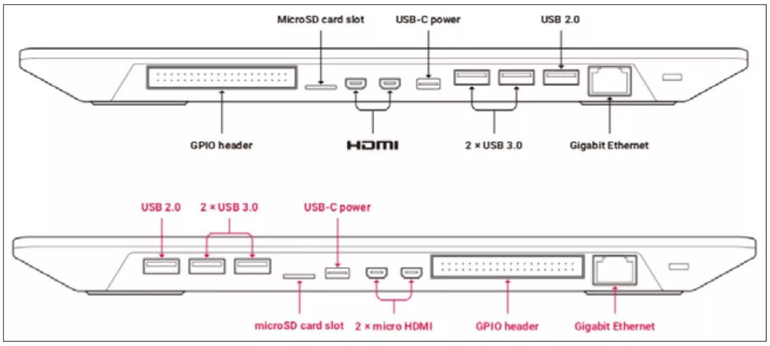

MicroSD card slot
USB-C power
USB 2.0
GPIO header
HDMI
2 × USB 3.0
Gigabit Ethernet
USB 2.0
2 × USB 3.0
USB-C power
microSD card slot
2 × micro HDMI
GPIO header
Gigabit Ethernet

Figure 3: Port repositioning between the Pi 400 [5] and Pi 500 [6] provides a choice of screen placement. Do you prefer the left, or the right side? © Raspberry PI

a Pimoroni HyperPixel 4-inch TFT display (Figure 5), on account of its higher resolution (800x480, 18-bit color, 60 frames per second), as the best option for a compact, high-contrast console screen with a display parallel interface (DPI). Mounting directly on the Cyberdeck HAT with four stand-offs and a GPIO extension (both included with the HyperPixel) requires careful and deliberate action.

Connecting the HyperPixel requires modifying the Cyberdeck HAT by slicing a jumper trace (Figure 6), disabling Stemma connector 18 [11]. Cutting the trace resolves a pin conflict between the display and the connector. Exercise care not to push through the other side into the screen (especially if using alternative stand-offs) or to warp the panel during assembly: It is a rather delicate part. Complete the board setup with an external HDMI monitor and proceed through the normal Raspberry Pi OS initialization steps to configure a user, WiFi access, and the like. Next, edit the file `/boot/firmware/config.txt` with root privileges to add the line

```
dtoverlay=vc4-kms-dpi-hyperpixel4,rotate=270
```

which configures screen output and adjusts console orientation. The X desktop requires separate alignment, most easily achieved by visiting *Preferences* | *Screen Configuration* | *Orientation* and selecting *left* (Figure 7). What's more, this tool can be used to configure multiple monitors if you choose to use one or both of the Pi's HDMI ports in addition to the Cyberdeck itself.

Warez

The limited resolution possible on a 4-inch screen limits what is

Table 1: Cyberpunk Thesaurus

Zaibatsu	Large industrial or corporate conglomerate
Deliverator	High-speed delivery driver for a mobster-run pizza chain
Arachnofiber	Futuristic fabric similar to kevlar
Deck	A hacker's computer, very 'leet
Slicing	Hacking, often said of breaking cryptographic codes
Flatline	Breaking. *He carelessly flatlined the TFT panel.*
Warez	Software. Usually the hacker's own toolkit.
Jack	Connecting, especially to the global Net.

The esoteric terminology used in this article originates in the novels of William Gibson and Neal Stephenson [9].

Table 2: Bill of Materials

Das Loot	Price (US$)	Source
Raspberry Pi 500	$97.22	[https://thepihut.com/products/raspberry-pi-500]
Sleeve for Raspberry Pi 500/400	$5.63	[https://thepihut.com/products/sleeve-for-raspberry-pi-400]
Raspberry Pi 27W USB-C power supply	$12.83	[https://thepihut.com/products/raspberry-pi-27w-usb-c-power-supply]
Adafruit Cyberdeck HAT	$8.95	[https://www.adafruit.com/product/4863]
Pimoroni HyperPixel	$54.95	[https://www.adafruit.com/product/3932]

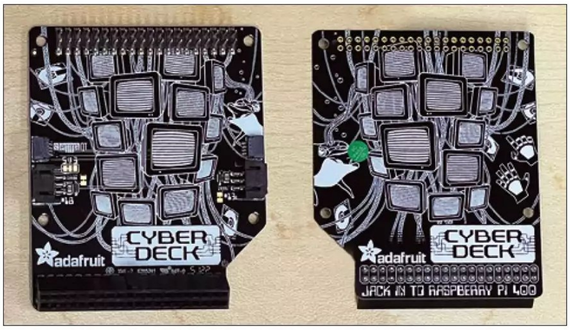

CYBER DECK
adafruit
CYBER DECK
adafruit
JACK IN TO RASPBERRY PI 400

Figure 4: The cyberdeck from Adafruit zaibatsu is the same hardware Kevin Mitnick used when he popped Sidewinder. Or not. If a product description was ever worth reading, it's this one's.

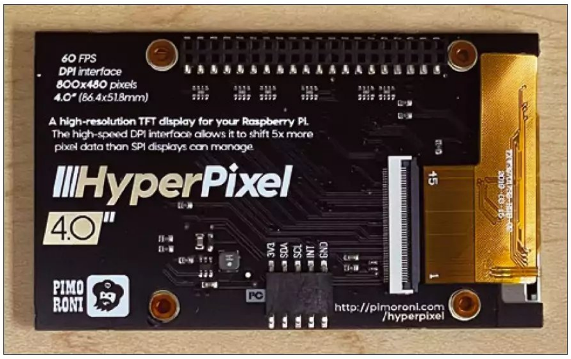

60 FPS
DPI interface
800x480 pixels
4.0" (86.4x51.8mm)
A high-resolution TFT display for your Raspberry Pi.
The high-speed DPI interface allows it to shift 5x more pixel data than SPI displays can manage
HyperPixel
4.0"
PIMO RONI
http://pimoroni.com/hyperpixel

Figure 5: The Pimoroni HyperPixel comes equipped with all necessary parts to complete the build.

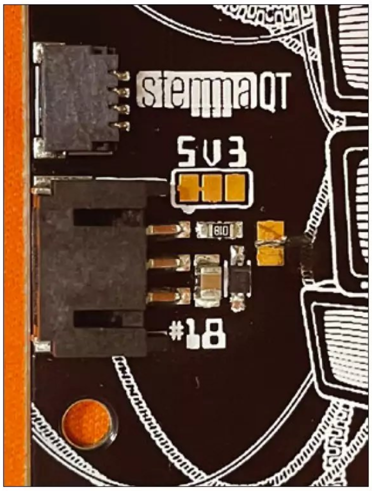


Figure 6: Slicing the jumper – not nearly as rewarding as slicing the crypto.

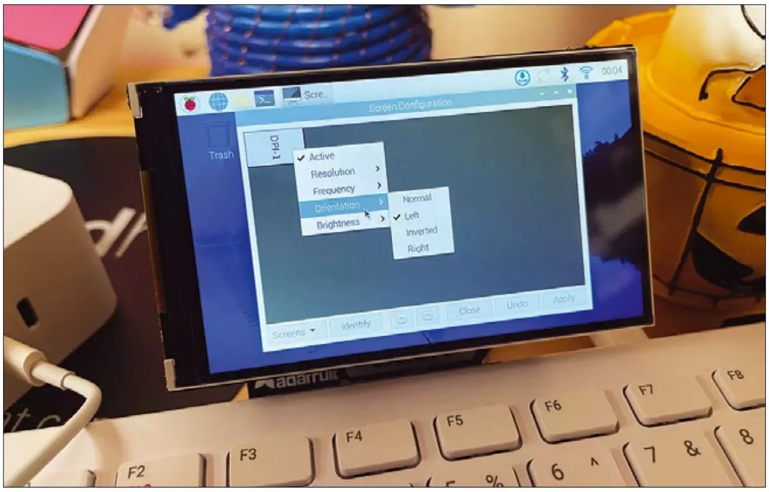


Figure 7: Setting the orientation for the graphical interface in the Raspberry Pi tools completes the setup. The Pimoroni HyperPixel mounts directly on the Cyberdeck HAT. Assemble with care, or you could entirely accidentally flatline the TFT like a certain careless author.

reasonably possible on the deck's GUI, which in any event lacks a mouse. Fortunately, I have explored terminal user interface (TUI) tools often in this column. Tools like `btop` [12], `glances` [13], `speedometer` [14], `nethogs` [15], and even plain old `top` [16] are ready to serve (Figure 8).

Now you can jack your Cyberdeck to the net, unleash your warez, and … hack the planet! [17] ■

Info

[1] Acer Aspire One: [https://en.wikipedia.org/wiki/Acer_Aspire_One]

[2] Rolling Rhino Ubuntu remix: [https://rollingrhino.org]

[3] Raspberry Pi 400: [https://www.raspberrypi.com/products/raspberry-pi-400/]

[4] Raspberry Pi 500: [https://www.raspberrypi.com/products/raspberry-pi-500/]

[5] *Raspberry Pi 400*, Raspberry Pi Ltd, January 2022, pg. 3: [https://datasheets.raspberrypi.com/rpi400/raspberry-pi-400-product-brief.pdf]

[6] *Raspberry Pi 500*, Raspberry Pi Ltd, December 2024, pg. 3: [https://datasheets.raspberrypi.com/pi500/raspberry-pi-500-product-brief.pdf]

[7] "Pushing Raspberry Pi Storage to Its Limit" by Federico Lucifredi, *ADMIN*, issue 84, 2024, pg. 91: [https://www.admin-magazine.com/Archive/2024/84/Pushing-Raspberry-Pi-storage-to-its-limit]

[8] Stephenson, Neal. *Snow Crash*. Spectra, 2003: [https://archive.org/details/SnowCrashNealStephenson]

[9] Stemma and Stemma QT: [https://learn.adafruit.com/introducing-adafruit-stemma-qt/what-is-stemma]

[10] Analog knobs on Raspberry Pi 400 with Cyberdeck HAT: [https://learn.adafruit.com/analog-knobs-on-raspberrypi-400-with-cyberdeck-hat]

[11] Fixing pin conflicts: [https://learn.adafruit.com/cyberdeck-bonnet-and-hat/fixing-pin-conflicts#option-1-cut-a-trace-3104511-6]

[12] "Bottom: The New top" by Federico Lucifredi, *ADMIN*, issue 80, 2024, pg. 94

[13] "System Temperature as a Dimension of Performance" by Federico Lucifredi, *ADMIN*, issue 72, 2022, pg. 93: [https://www.admin-magazine.com/Archive/2022/72/System-temperature-as-a-dimension-of-performance]

[14] "Filesystem Murder Mystery" by Federico Lucifredi, *ADMIN*, issue 50, 2019, pg. 94: [https://www.admin-magazine.com/Archive/2019/50/Filesystem-Murder-Mystery]

[15] nethogs(8) man page: [https://manpages.ubuntu.com/manpages/bionic/en/man8/nethogs.8.html]

[16] top(1) man page: [https://manpages.ubuntu.com/manpages/focal/en/man1/top.1.html]

[17] Rafael Moreu and Iain Softley, Hackers: [https://www.imdb.com/title/tt0113243/]

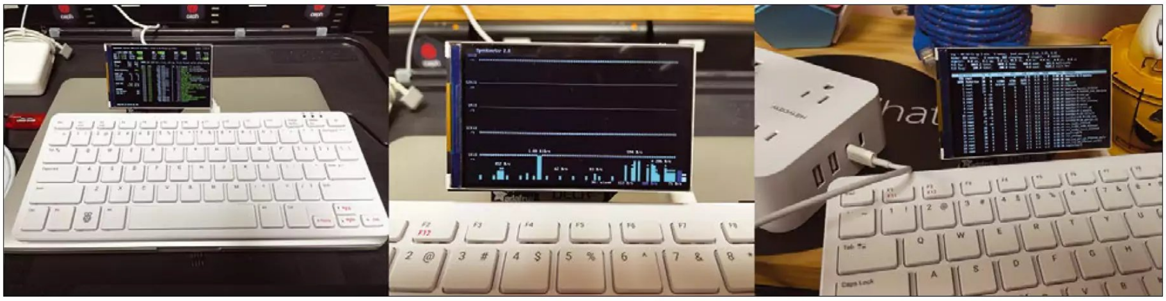
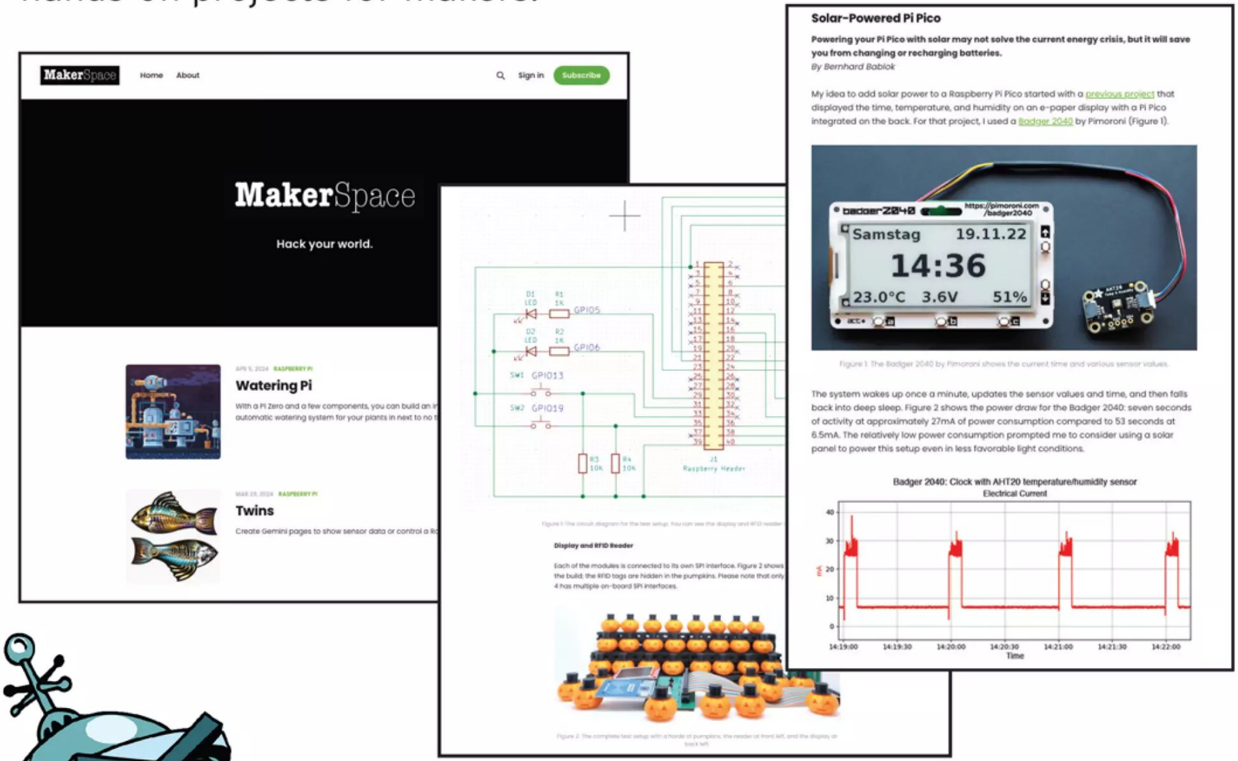


Figure 8: My shiny new deck in action. From the left `glances`, `speedometer`, and `top`.

MakerSpace-Online

New Maker Content Every Week

At MakerSpace, we are all about technology you can use to build your own stuff. Our goal is to help you turn your ideas into reality with hands-on projects for makers.



MakerSpace
Home
About
Sign in
Subscribe
MakerSpace
Hack your world.
RASPBERRY PI
RASPBERRY PI
D1 LED
R1 1K
GPIO5
D2 LED
R2 1K
GPIO6
SW1 GPIO13
SW2 GPIO19
R3 10K
R4 10K
J1 Raspberry Header
badger2040
https://pimoroni.com/badger2040
Samstag 19.11.22
14:36
23.0°C 3.6V 51%
Badger 2040: Clock with AHT20 temperature/humidity sensor
Electrical Current
mA
0
10
20
30
40
14:19:00
14:19:30
14:20:00
14:20:30
14:21:00
14:21:30
14:22:00
Time

Watering Pi

With a Pi Zero and a few components, you can build an automatic watering system for your plants in next to no

Twins

Create Gemini pages to show sensor data or control a Ra

Display and RFID Reader

Each of the modules is connected to its own SPI interface. Figure 2 shows the build; the RFID tags are hidden in the pumpkins. Please note that only 4 has multiple on-board SPI interfaces.

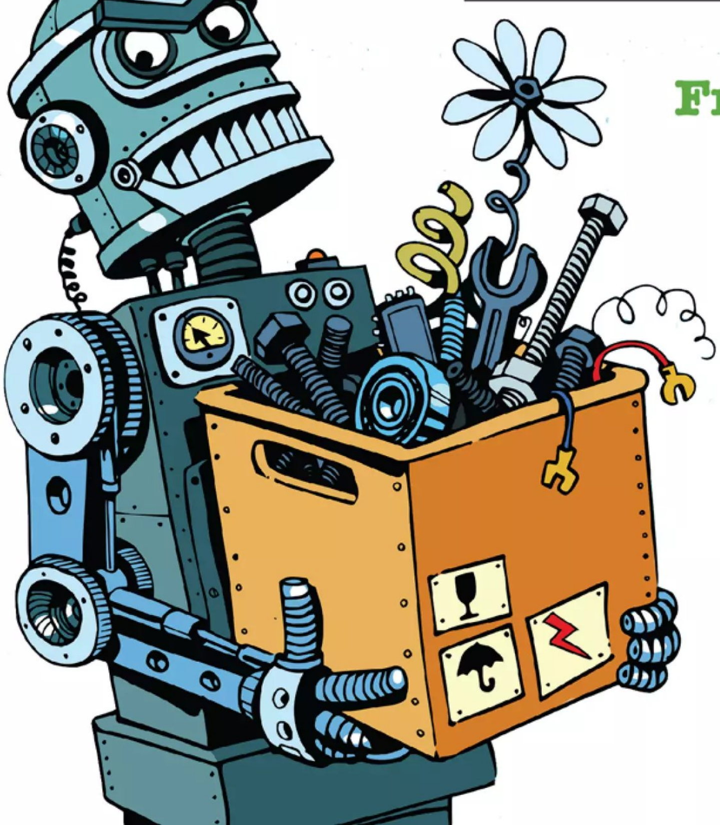
Solar-Powered Pi Pico

Powering your Pi Pico with solar may not solve the current energy crisis, but it will save you from changing or recharging batteries.
By Bernhard Bablok

My idea to add solar power to a Raspberry Pi Pico started with a previous project that displayed the time, temperature, and humidity on an e-paper display with a Pi Pico integrated on the back. For that project, I used a Badger 2040 by Pimoroni (Figure 1).

Figure 1: The Badger 2040 by Pimoroni shows the current time and various sensor values.

The system wakes up once a minute, updates the sensor values and time, and then falls back into deep sleep. Figure 2 shows the power draw for the Badger 2040: seven seconds of activity at approximately 27mA of power consumption compared to 53 seconds at 6.5mA. The relatively low power consumption prompted me to consider using a solar panel to power this setup even in less favorable light conditions.



Fresh Content, Delivered

Subscribe now and join the MakerSpace community. You'll stay up-to-date when new content is published.

Join Now!

https://makerspace-online.com

ADMIN
Network & Security

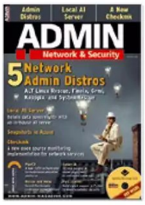
NEWSSTAND

Order online:
bit.ly/ADMIN-Library

ADMIN is your source for technical solutions to real-world problems. Every issue is packed with practical articles on the topics you need, such as: security, cloud computing, DevOps, HPC, storage, and more! Explore our full catalog of back issues for specific topics or to complete your collection.

#88 – July/August 2025

5 Network Admin Distros

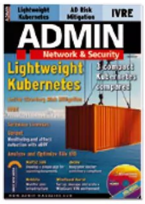


Admin distros take both workstations and servers into account, have broad support for various filesystems, deploy in heterogeneous environments without restrictions, and come with the necessary tool collections.

On the DVD: openSUSE Leap 15.6

#87 – May/June 2025

Lightweight Kubernetes

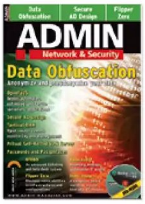


K3s, k0s, and MicroK8s vie for performance honors on the control plane and data plane in artificially created extreme stress scenarios.

On the DVD: AlmaLinux 9.5 Minimal

#86 – March/April 2025

Data Obfuscation

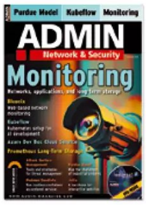


Generalization, suppression, perturbation, and differential privacy are essential data protection techniques that enable a balance between data security and usability and ensure compliance with legal requirements.

On the DVD: Rocky Linux 9.5 Minimal

#85 – January/February 2025

Monitoring

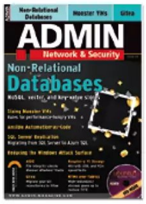


We look at infrastructure and systems monitoring and evaluate the tools needed to scale or store data in the long term for analysis.

On the DVD: Fedora Server 41

#84 – November/December 2024

Non-Relational Databases

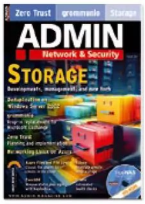


NoSQL databases manage data unsuitable for table-based relational databases and deliver efficiencies in scaling and high availability.

On the DVD: Ubuntu 24.04 LTS Server "Noble Numbat"

#83 – September/October 2024

Storage



Improved management, suitable drivers, standardized protocol structures, and advancements in future-proof hardware and software lead to more durable, more manageable, and easier-to-repair storage.

On the DVD: TrueNAS SCALE 24.04 "Dragonfish"

WRITE FOR US

Admin: Network and Security is looking for good, practical articles on system administration topics. We love to hear from IT professionals who have discovered innovative tools or techniques for solving real-world problems.

Tell us about your favorite:

- Interoperability solutions
- Practical tools for cloud environments
- Security problems and how you solved them
- Ingenious custom scripts
- Unheralded open source utilities
- Windows networking techniques that aren't explained (or aren't explained well) in the standard documentation

We need concrete, fully developed solutions: installation steps, configuration files, examples – we are looking for a complete discussion, not just a "hot tip" that leaves the details to the reader.

If you have an idea for an article, send a 1-2 paragraph proposal describing your topic to: *edit@admin-magazine.com*.

Authors

Amber Ankerholz	6
Dr. Lothar Burger	71
Conor Fitzgerald	50
Marcin Gastol	62
Tam Hanna	36
Ken Hess	3
Thomas Joos	80
Heike Jurzik	86
Philip Lorenz	10, 40, 76
Martin Gerhard Loschwitz	14, 26, 44, 66
Federico Lucifredi	92
Markus Stubbig	20, 32, 56
Matthias Wübbeling	54

Contact Info

Editor in Chief
Joe Casad, jcasad@linuxnewmedia.com

Managing Editors
Rita L Sooby, rsooby@linuxnewmedia.com
Lori White, lwhite@linuxnewmedia.com

Senior Editor
Ken Hess

Localization & Translation
Ian Travis

News Editor
Amber Ankerholz

Copy Editors
Amy Pettle, Aubrey Vaughn

Layout
Dena Friesen, Lori White

Cover Design
Dena Friesen, Illustration based on graphics by vska, 123RF.com

Advertising
Brian Osborn, bosborn@linuxnewmedia.com

Publisher
Brian Osborn

Marketing Communications
Gwen Clark, gclark@linuxnewmedia.com
Linux New Media USA, LLC
4840 Bob Billings Parkway, Ste 104
Lawrence, KS 66049 USA

Customer Service / Subscription
For USA and Canada:
Email: cs@linuxnewmedia.com
Phone: 1-785-856-3080

For all other countries:
Email: subs@linuxnewmedia.com
www.admin-magazine.com

While every care has been taken in the content of the magazine, the publishers cannot be held responsible for the accuracy of the information contained within it or any consequences arising from the use of it. The use of the DVD provided with the magazine or any material provided on it is at your own risk.

Copyright and Trademarks © 2025 Linux New Media USA, LLC.

No material may be reproduced in any form whatsoever in whole or in part without the written permission of the publishers. It is assumed that all correspondence sent, for example, letters, email, faxes, photographs, articles, drawings, are supplied for publication or license to third parties on a non-exclusive worldwide basis by Linux New Media unless otherwise stated in writing.

All brand or product names are trademarks of their respective owners. Contact us if we haven't credited your copyright; we will always correct any oversight.

Printed in Nuremberg, Germany by be1druckt GmbH.

Distributed by Seymour Distribution Ltd, United Kingdom

ADMIN (Print ISSN: 2045-0702, Online ISSN: 2831-9583, USPS No: 347-931) is published bimonthly by Linux New Media USA, LLC, and distributed in the USA by Asendia USA, 701 Ashland Ave, Folcroft PA. September/October 2025. Application to Mail at Periodicals Postage Prices is pending at Philadelphia, PA and additional mailing offices. POSTMASTER: send address changes to Linux Magazine, 4840 Bob Billings Parkway, Ste 104, Lawrence, KS 66049, USA.

Represented in Europe and other territories by: Sparkhaus Media GmbH, Bialasstr. 1a, 85625 Glonn, Germany.

ADMIN 90
Available Starting
December 5

Our next issue will be packed with all the great content you expect from *ADMIN*. Here are a few of the upcoming articles:

- **MLOps Pipeline**
- **Hotpatching in Win Server 2025**
- **Kata Containers**
- **JVM Memory Management**
- And much more!

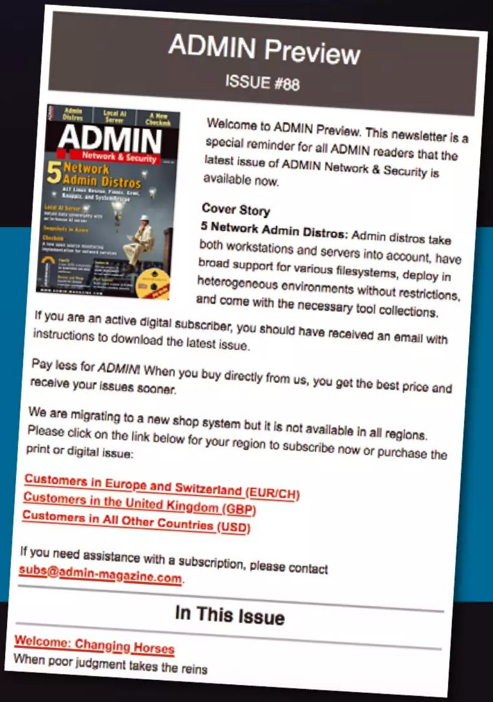
Please note: Articles could change before the next issue.

BE THE FIRST TO SEE WHAT'S NEXT

Subscribe free to the *ADMIN* Preview newsletter and get a sneak peek at every article included in the next issue of *ADMIN*.

Sign up today at https://bit.ly/admin-preview

Image © artnovielysa, 123RF.com



ADMIN Preview

ISSUE #88

Welcome to ADMIN Preview. This newsletter is a special reminder for all ADMIN readers that the latest issue of ADMIN Network & Security is available now.

Cover Story

5 Network Admin Distros: Admin distros take both workstations and servers into account, have broad support for various filesystems, deploy in heterogeneous environments without restrictions, and come with the necessary tool collections.

If you are an active digital subscriber, you should have received an email with instructions to download the latest issue.

Pay less for *ADMIN*! When you buy directly from us, you get the best price and receive your issues sooner.

We are migrating to a new shop system but it is not available in all regions. Please click on the link below for your region to subscribe now or purchase the print or digital issue:

Customers in Europe and Switzerland (EUR/CH)
Customers in the United Kingdom (GBP)
Customers in All Other Countries (USD)

If you need assistance with a subscription, please contact subs@admin-magazine.com.

In This Issue

Welcome: Changing Horses
When poor judgment takes the reins

SFSCON
FREE SOFTWARE CONFERENCE

NOI
TECHPARK SÜDTIROL / ALTO ADIGE

NOV
7TH
8TH

2025

BE PART OF IT!

2 INSPIRING DAYS
1 FESTIVAL
100 SPEAKERS

SFSCON.IT

NOI TECHPARK
BOLZANO/BOZEN

GRUPPO FOS soluzioni ad alta tecnologia
TELMEKOM NETWORKS
VATES Open Infrastructure made simple
alpitronic
MADE IN CIMA stunning websites that work
pandigital
Zirkonzahn
1006.org
{catchsolve}
Christian Gapp
deda.next ENVISION PUBLIC SERVICES
peer
RMB tec
SiMedia your perfect guest.
Südtirol Business School
SUGGESTO SMART experience

NOI TECHPARK
SÜDTIROL / ALTO ADIGE

A.-VOLTA-STRASSE 13A, BOZEN
VIA A. VOLTA 13A, BOLZANO

VISIT US ON
NOI.BZ.IT

HETZNER

We really got your back!

Hetzner: As secure as your home setup – minus the cable spaghetti.

We have the tools every sysadmin dreams of:

- ✓ **RustDesk » *One-Click App***
  Your own remote desktop server. No third-party provider. No hidden services.
  Full control – in seconds.
- ✓ **WireGuard » *One-Click App***
  Your ultra-light VPN server – set up in no time, powered by modern cryptography.
- ✓ **Storage Box**
  Reliable storage. Redundant and RAID-secured. Perfect for your backups.

TRY HETZNER CLOUD
FOR ONE MONTH
FOR FREE

HetznerXAdmin10

Redeem code & save $ 10

htznr.li/admin/4sysadmins

Credit code worth $10 / €10; for new customers; activate before 06/30/26; valid for 3 months after activation. Errors excepted. All rights reserved by the respective manufacturers.